

Productivity Pak III

RE RUN

C-64/128 Applications

Word Processor

Database

Spreadsheet

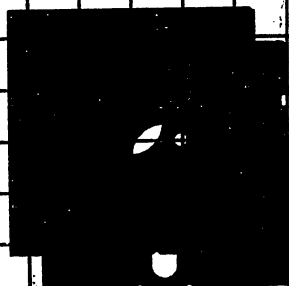
Notepad

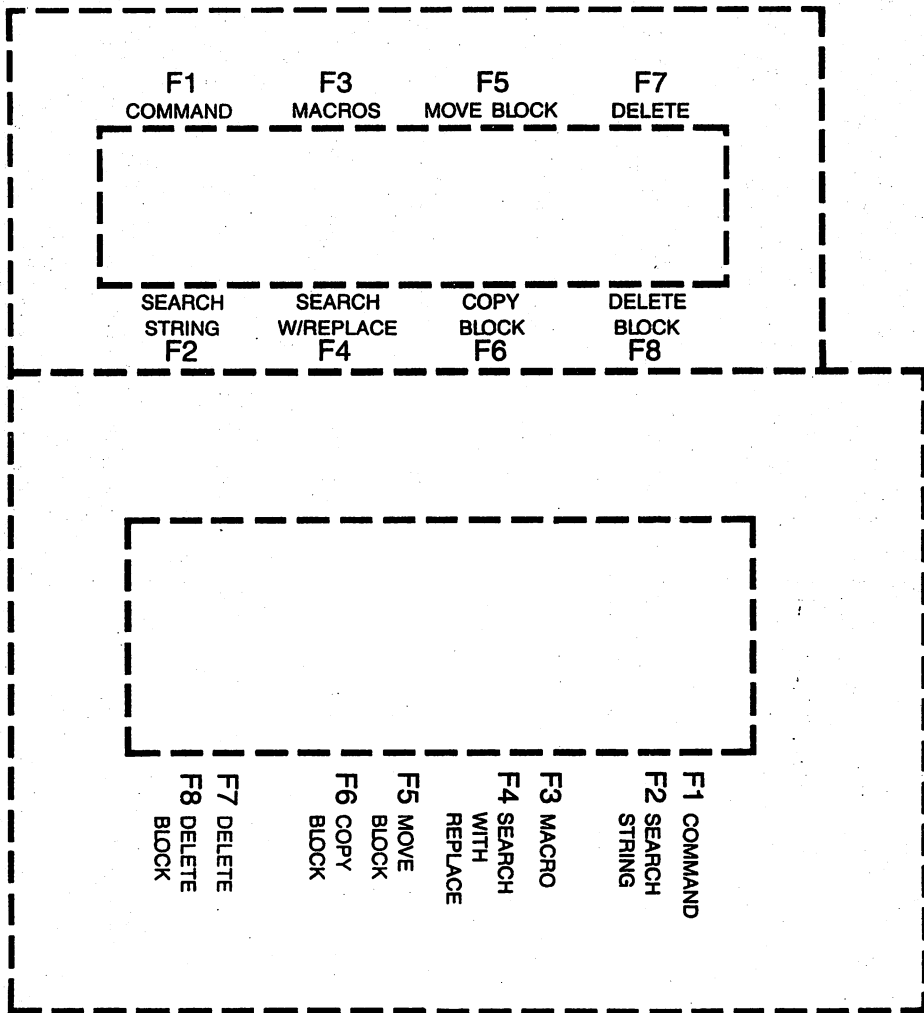
Electronic Index

Stock Portfolio^{**}

What You Need
On One Disk

or C-64 only
or C-128 only





Keyboard overlays for RUN Script Plus.

Introduction

Welcome to RUN'S Productivity Pak III, 1987

This year has proven notable for the appearance of new productive software for both the C-64 and C-128. New word processors, databases and application programs are replacing arcade games and flight simulators as the programs of choice among many users. As evidence, take a look at the lineup of C-64 and C-128 programs we've readied for the Productivity Pak III. So many good application programs were available that we were able to devote one side of the disk exclusively to 64 and the other to 128 programs.

First and foremost on the list is RUN Script Plus. Updated and overhauled especially for Productivity Pak III, this time-honored word processor is in fine form to lure C-64 and C-128 users into enjoying its remarkable options. A spelling checker, dictionary maintenance program and a user-expandable dictionary round out its repertoire of goodies.

Next, we offer RUN File, another program written especially for Productivity Pak III. An all-new database for collecting, organizing and printing records, RUN File looks like a sure winner. Written in both 64 and 128 versions by the renowned Commodore graphics programmer, David Darus, RUN File places recipes, receipts and reminder notes of all sorts into an easy-to-access for-

mat for your quick reference.

A first for the Productivity Pak series is a 64 and 128 spreadsheet, aptly named RUN Calc. The 64 version, written especially for this issue by spreadsheet programmer Trent Busch, offers an effective design that makes it smooth and fast in operation. RUN Calc 128, a beginner's program, should prove handy for learning spreadsheet fundamentals.

We also present both the 64 and 128 versions of Bob Kodadek's Notepad series of programs. Surely, any programmer will find uses for these utilities. They're handy, simple to use and versatile.

Commodore-128 users interested in stock market investments will appreciate RUN Investor, a program designed to maintain your stock portfolio. Finally, RUN Dex, written entirely in machine language, finds its niche in the Productivity Pak III as a C-64 filing program that lets you create lists of names and addresses, telephone numbers and other data.

That just about sums up this year's Productivity Pak III. With such an impressive list of programs, my only concern now is how to improve on it for next year!

Tim Walsh
Technical Editor
RUN magazine

Directory

Page	Article	Disk Filename	File Type
C-128 PROGRAMS			
1	RUN Script	RS128	BASIC
		CS.ROUND3	ML
		OB.RS NMI 2.5	ML
		OB.RS128 2.56	ML
	(Dictionary Maintenance)	RS128 DICT MAINT	BASIC
		OB.RS128 DMAINT	ML
	(Spell Checker)	RS128 SPELLER	BASIC
		OB.RS128 SPELLER	ML
20	RUN File 1.0	RUNFILE 128	COMPILED
		RUNFILE 128.BAS	BASIC
25	RUN Calc 128	RUNCALC 128	BASIC
41	RUN Notepad 128	NOTEPAD 128	BASIC
		128 NOTEPAD.OBJ	ML
50	RUN Investor	RUN INVESTOR 128	BASIC
C-64 PROGRAMS			
1	RUN Script	RS64	BASIC
		OB.RS64 2.55	ML
	(Dictionary Maintenance)	RS64 DICT MAINT	ML
	(Spell Checker)	RS64 SPELLER	ML
20	RUN File 1.0	RUNFILE 64	COMPILED
		RUNFILE 64.BAS	BASIC
37	RUN Calc 64	RUNCALC 64	COMPILED
		RUNCALC 64.BAS	BASIC
45	RUN Notepad 64	RUN NOTEPAD 64	BASIC
		64 NOTEPAD V3.0	ML
47	RUN Dex	RUN MEMO BOOK	ML

How To Load

Loading from Menu

To get started, C-64 users should type LOAD "MENU 64",8 and press the return key. When you get the Ready prompt, the menu is loaded and you should type RUN to see a list of the programs on your disk. C-128 users need only press the shift and run-stop keys. When all the programs are displayed on the screen, you can run the one you select by pressing a single key.

Loading from Keyboard

If you do not wish to use the menu program, follow these instructions:

C-64:

To load a C-64 program written in Basic, type:

LOAD "DISK FILENAME",8

and then press the return key. The drive will whirl while the screen prints LOADING and then READY, with a flashing cursor beneath. Type RUN and press the return key. The program will then start running.

To load a C-64 program written in machine language (ML), type:

LOAD "DISK FILENAME",8,1

C-128:

All C-64 programs can be run on the C-128 as long as your computer is in C-64 mode.

All C-128-mode programs are clearly labeled on the directory page. Your C-128 *must* be in C-128 mode to run these programs.

To load a C-128-mode program, press the F2 key, type the disk filename and then press the return key. When the program has loaded, type RUN.

Making Copies of ReRUN Disks

Many of the programs on your ReRUN disk have routines that require you to have a separate disk onto which the program writes or saves subfiles. In order for you to use these programs, you will first have to make a copy of the original program onto another disk that has enough free space on it to hold these newly written subfiles.

If the program is written in Basic, it is simple to make a copy of the program. Just load the program into your computer following the procedures outlined above, and then save the program back onto a separate disk that has plenty of free space for extra files.

If the program is written in ML, copying is not so simple. You cannot simply load and save an ML program. In this case, you'll need to use a disk-backup utility program, such as the one on your Commodore Test Demo disk.

ReRUN Staff

Technical Editor: *Tim Walsh*

Managing Editor/Production: *Swain Pratt*

Copy Editor: *Peg LePage*

Proofreader: *Harold Bjornsen*

Design and Layout: *Anne Dillon*

Typesetting: *Doreen Means, Beth Krommes, Ken Sutcliffe*

Special Products Director: *Paul Finch*

Special Products Assistants: *Debbie Bourgault, Robyn Johnson*

Direct Marketing Coordinator: *Debbie Walsh*

RUN Script Plus

By Robert Rockefeller

RUN It Right

C-64 or C-128; disk drive; printer

RUN Script 64 & 128 have been substantially upgraded for Productivity Pak III. New features include a spelling checker equipped with a dictionary-maintenance utility; label-printing and RAM-expander capabilities; and the ability to load files and issue disk commands from a disk-directory listing.

The first part of the following documentation reviews the operation, commands and features of the main RUN Script Plus word processing program. Those readers who are already familiar with RUN Script may wish to proceed to the section on the new commands and, following that, to the description of the spelling checker and dictionary.

LOADING PROCEDURE

C-64 users must load and run RS64, a Basic program that automatically loads the RUN Script Plus machine language program. Similarly, C-128 users must load and run RS128.

As you're reading this documentation, keep in mind that when two

names of keys are separated by a slash, the keys should be pressed simultaneously. When they are separated by a comma, the keys should be pressed one after the other.

EDITING FUNCTIONS

RUN Script Plus offers many editing functions, which I tried to keep as intuitive as possible.

RUN Script Plus uses one line at the top of the screen to display prompts, operation and error messages. This leaves 24 lines free for displaying text. About 34,000 bytes are available for storing text in the C-64 version, and about 65,000 bytes in the C-128 version.

ABORTING OPERATIONS

You may abort almost any operation, including saving, printing and loading text, by pressing the CTRL key along with the Commodore key. About the only operation you can't stop is the replace-string function, once replacing has begun; but pressing the CTRL/Commodore combination during the input stage of this function will abort it.

In RUN Script Plus, the restore key aborts operations but has been modified from earlier versions of RUN

Script. See the section marked \$ under New RUN Script commands.

MOVING THE CURSOR

In RUN Script Plus, seven keys and key combinations function as cursor controls. These include the four cursor keys (eight on the C-128), the home key, the CTRL/back-arrow combination and the shift/return combination. To use the CTRL/back-arrow, press the CTRL and back-arrow keys together; likewise, press the shift and return keys simultaneously for shift/return.

Cursor keys—By pressing the cursor-down key or the (shifted) cursor-up key, you can scroll vertically through the text area. By pressing the cursor-right-and-left key, you can move the cursor horizontally.

With RUN Script Plus's word-wrap feature, a given screen line may have anywhere from one to 79 spaces at the end of it. The empty spaces appear as small dots that don't exist in the text area in memory, but are printed to the screen to pad out the line. When moving the cursor horizontally, you'll find it will skip over these small dots.

Shift/return—This key combination moves the cursor to the start of the next line.

Home—Pressing the home key once moves the cursor to the upper-left corner of the screen. Pressing it twice moves the cursor to the start of the manuscript.

CTRL/back-arrow—Pressing the CTRL key with the back-arrow key moves the cursor to the bottom-left corner of the screen, and pressing

this combination twice moves the cursor to the end of the text.

INSERTING TEXT

There are three ways to insert text with RUN Script Plus:

Shift/INST—Simultaneously pressing the shift key and the INST key inserts one space at the position of the cursor.

CTRL/I—Pressing the CTRL key and the I key toggles you in and out of Insert mode. Whenever you press a key in this mode, a character is inserted at the cursor position.

Run-stop—When you need to insert text at the start of a long document, press the run-stop key to insert a block of 200 spaces. To insert more than 200 spaces, press the run-stop key continuously until the insert block is large enough to suit your needs.

Shift/run-stop—This key deletes a block of inserted spaces. Just position the cursor at the beginning of the block and press shift/run-stop.

DELETING TEXT

The four methods of deleting text with RUN Script Plus are as follows:

DEL—Pressing the DEL key deletes the character to the left of the cursor.

F7—Pressing F7 also deletes the character *under* the cursor, and the cursor does not move.

F8—Pressing F8 deletes a block of text. Position the cursor over the first character of the section of text you wish to delete and press F8. A "delete block" message will appear on the status line. Then move the

cursor to the *last* character of the block you want deleted and press the return key. The block of text will be removed, but not yet lost, in case you change your mind.

The deleted block is copied to a buffer at the top of memory. To retrieve the block, press F6 twice. A "select insert point" message will then appear on the status line. Move the cursor to where you want the deleted text to appear and press the return key again.

F8,F8—Pressing F8,F8 will delete all text from the cursor position to the end of the text area. You will be prompted to answer y/n before erasing is performed. You can abort both F8 key functions without deleting any text by simultaneously pressing the CTRL and Commodore keys.

MOVING AND COPYING TEXT

F5—To move a block of text, position the cursor over the first character of the block and press F5. A "move block" message will appear. Then move the cursor to the last character of the block and hit the return key. The text will be deleted and a "select insert point" message will now appear. Position the cursor where you want the text inserted, press the return key, and the block of text will appear in the new position.

F6—To copy a portion of the text in memory to another position, press F6 once after positioning the cursor over the first character of the block to be copied. A "copy block message" will appear on the status line. Next, move the cursor to the last character of the block to be copied

and hit the return key. A "select insert point" message will be displayed. Move the cursor to the position where you want the block inserted, hit the return key, and the block will appear in the new location.

F6,F6—To insert the contents of the text buffer into your document, press F6 twice. After a "select insert point" message appears, move the cursor to the insert point and press the return key. The contents of the buffer will be inserted. This feature allows you to make multiple copies of the same text block or to recover a deleted block as explained above.

SEARCH AND SEARCH-AND-REPLACE

F2—To search for a string between the cursor position and the end of the text, press F2 once and a "search string?" prompt will appear. Type in the string you wish to find, then press the return key. If a match is found, the cursor will appear over the match. If no match is found, a "string not found" message will be displayed. After entering a search string, you can search for multiple occurrences of the string by pressing F2 twice in rapid succession.

F4—Use F4 to replace strings within the text area. A search string and replace string will be requested as above, but at each occurrence of the search string, the string will be displayed so you can elect to replace it or bypass it.

F4,F4—This prompts you to input a search string and a replace string. After you've done this, all the occurrences of the search string from

the cursor position to the end of text will be replaced by the replace string.

CHANGING DISPLAY COLORS

To change the colors of the RUN Script Plus screen display, use the following key combinations:

CTRL/1—to change the text color.

CTRL/2—to change the background color. Changes C-128 screen color.

CTRL/3—to change the color of the status line.

CTRL/4—to change the border color (C-64 version only).

OTHER EDIT FUNCTIONS

Return—Pressing the return key when entering text signals the end of a paragraph.

CTRL/x—This combination interchanges the two characters to the left of the cursor—a fast method of correcting transposed letters.

CTRL/6—This combination toggles RUN Script Plus in and out of Shift Lock mode. When the **""cap""** message is displayed on the status line, all alphabetic characters typed will be capitalized.

CTRL/=—If you should want to replace the line-padding dots with some other character, press the CTRL and equals-sign keys simultaneously. A **""chr""** message will appear on the status line. Then type the replacement character, and it will replace the dots. Once this is done, you cannot retrieve the dots.

HELP—(C-128 version only). Moves the cursor to the other text area when the display is split. If it's

in area 1, it'll move to area 2, and vice versa.

CTRL/z—Prints five spaces to the screen. This function is useful for indenting paragraphs.

CTRL/u—Enables you to select a new character to be inserted into the text area when you press the run-stop key. Type CTRL/u, then any character. The default character is the space.

F1—Pressing the F1 key after a prompt for a filename activates a screen-read feature.

Somewhere at the start of a document, within the first 256 characters, create a comment with the .cm dot command. Following the .cm command, type the document's filename surrounded by double quotes, such as .cm "0:filename".

When you save the file with the F1,@ or F1,s command, if you press the F1 key after the "filename ?" prompt appears, RUN Script Plus will read the filename from the screen into the input line. This saves you from having to type the filename every time.

THE F1 KEY

When you press F1, a "command ?" prompt will appear. You then enter the letter corresponding to the function (as listed below) that you wish to execute.

d—F1,d selects the device with which all saving and loading will take place. This may be device 8 or 9, the disk drive. The default device is number 8.

s—F1,s saves text to the selected device.

@—F1,@ saves with replace. The old file is automatically scratched before the new file is saved.

I—F1,I loads a text file from the disk.

a—F1,a is an append function for combining a text file on disk with the one in memory. The file coming from disk is appended starting at the cursor position and replaces any text that follows the cursor. (This append function is not intended for use with sequential files.)

t—F1,t selects whether characters will be output and input in ASCII or Commodore ASCII. The default mode is Commodore ASCII.

w—F1,w writes the text in memory to disk as either a sequential or a user file. If you want to save it as a sequential file, you must append the string ,s,w to the filename when entering the filename at the prompt. If you want to save the text as a user file, it's necessary to append ,u,w to the filename.

With the w function, you can write the text to disk using either Commodore ASCII or true ASCII, depending on how RUN Script Plus is configured. (See the t function, above.) Also, if you change file types between a read and a write, you need to rename the file. Finally, you can do a save-with-replace by inserting @0: before the filename. This is not recommended with the 1541, because of its DOS bug.

r—F1,r loads a sequential or user file from disk. Just enter the filename at the prompt.

\$—F1,\$ lists the disk directory. The \$ command has been modified for RUN Script Plus. See "New RUN

Script Commands."

> or <—F1,> issues a disk drive command, while **F1,<** reads the disk error channel.

f—F1,f reports the number of free bytes remaining.

x—F1,x exits to Basic.

p—F1,p prints text. After selecting the print function, you'll be asked how many copies to make. You can choose up to 99. Next, you'll be asked to select the output device. Output may be directed to device 3, the screen; device 4 or 5, the printer; or device 8 or 9, the disk drive.

If you select the printer or screen as the output device, you can output one page at a time. Simply answer n for no at the "continuous (y/n) ?" prompt. A "next output (c/p/s) ?" prompt will then be displayed before each page is output.

If you enter s for screen at this point, the next page will be printed to the screen. If you enter p for printer, the next page will be sent to your printer. If you enter c at the prompt, output from that point onward will be continuous, and the "next output (c/p/s) ?" prompt will no longer appear.

0—F1,0 loads a new character set. In Special Graphics mode, the business-graphics characters of this set would be printed using the printer's Dot Graphics mode.

1-9—F1,1 loads a new print set number 1. F1,2 through F1,6 load new print set numbers 2-6. F1,7 through F1,9 load new print set numbers 7-9, but work only in the C-128 version.

Cursor-up and cursor-down—F1

with the vertical cursor controls selects the Fast Scroll mode, which rapidly moves the text up or down. You leave this mode by pressing any other key, or abort it with the Commodore and CTRL keys.

Cursor-left and cursor-right—F1 with the horizontal cursor controls selects the Page mode. In this mode, pressing the cursor-right and cursor-left keys scrolls the text up and down 24 lines, respectively. Exit this mode by pressing any other key, or abort it by simultaneously pressing the Commodore and CTRL keys.

c—F1,c changes the case of all text from the cursor position to the document end. Uppercase letters become lowercase, and vice versa.

R—F1,R saves a copy of the table of redefined characters to the current save device. First, redefine characters with the .dc command (see the dot commands, below); then print the file to allow the .dc commands to execute; then save the table. Now you can load the saved characters at any time with the .lr command.

o—F1,o prints a document with one copy, output to printer device 4 and continuous output.

Z—F1,Z (C-128 version only) swaps text memory in RUN Script Plus's text area in bank 1 with the memory in bank 0. Pressing F1,Z again restores the original text. You can keep one text file in bank 1, one in bank 0, and switch between them with F1,Z. This permits up to 65,000 bytes of text to be stored in memory.

M—F1,M selects the macro characters that should be turned off before printing the left margin, a

header or a footer. This prevents problems such as underlining in the left margin. RUN Script Plus expects the same macro letter to be used to activate and deactivate printer functions. For example, if underlining is turned on with uppercase U, a lowercase u is needed to cancel it, and vice versa.

At the prompt, just enter the macro characters—up to 13 of them—that need to be cancelled. Only enter the macros that activate functions.

m—F1,m loads a set of macros as defined with the Define Macros program (see discussion below). If you're using a C-128, you must run Define Macros in C-64 mode, not 128 mode. However, the macros table it creates can be used with the C-128 version of RUN Script Plus.

g—F1,g enables and disables the Special Graphics mode.

G—F1,G selects the type of printer. If you have an MPS-803 or compatible, MPS-801, 1525 or MPS-1000 (Commodore mode), you answer yes at the prompt. Answer no if you don't have one of these printers. This function determines how RUN Script Plus will print dot-graphics characters.

C—F1,C selects the secondary address used to print dot graphics and sends macro command strings to the printer. Unlike the .ca dot command, this function changes the permanent secondary address inside RUN Script Plus.

S—F1,S splits the RUN Script Plus text display into two separate areas. Type F1 then S, and the prompt

"how many text areas (1/2) ?" will appear. If you answer 1, RUN Script Plus will remain unchanged.

If you enter 2, the prompt "size of area two ?" will appear. You may choose any number from 1 to 18, and text area 2 will be allocated that many kilobytes of memory. Since all of the current text is erased before the splitting occurs, there is a third prompt in this function, "erase text (y/n) ?". If you enter y for yes, the text area will split, and the cursor will appear at the start of text area 1.

A—F1,A provides movement between the two text areas. At the prompt, enter either 1 or 2, depending on which area you wish to access.

T—F1,T selects the secondary address used to print text to the printer. Unlike the .ta dot command, this function changes the permanent secondary address inside RUN Script Plus.

L—F1,L specifies whether a line-feed character should be printed after each carriage-return character. Unlike the .lf dot command, this function changes a permanent flag inside of RUN Script Plus.

q—F1,q saves a copy of RUN Script Plus, including any printer macros, the current color settings, the current settings from the F1,S, F1,T, F1,G, F1,C, F1,L, F1,M, F1,d, F1,t, F1,g, CTRL/= and CTRL/u commands, and the settings from the .gc, .gb, .ge, .bs and .ff dot commands. To customize RUN Script Plus, use these commands to configure it, then save it with F1,q.

DEFINING PRINTER MACROS

The printer-macro feature of RUN Script Plus enables you to customize your copy of this word processor so you can take full advantage of whatever capabilities your printer may possess.

An idiosyncrasy of RUN Script Plus is that the printer must be turned on when output to the screen is taking place, because a file is always opened to the printer when you select output to the screen. If RUN Script Plus ever seems to "hang up" mysteriously during a printout, check your printer.

You may select any upper- or lowercase alphabetic character to be a macro character. You then create a table of printer macros with the Define Macros program. Each macro character represents a string of user-defined characters. When a macro character is encountered during printing, this string, rather than the macro character itself, will be sent to the output device.

For example, let's say you own a printer that requires the sequence ESC X (decimal values 27 and 88) to start printing double-width characters. With Define Macros, you can select a character—D, for instance—to represent this two-character string. Then, when D is encountered during printing, the decimal sequence 27,88 will be sent to the printer to produce double-width characters. You could define another character, perhaps d, to represent the sequence to stop printing double-width characters.

To create a double-width heading,

first place the cursor in front of the heading, then press the F3 key. A "***mac*" message will appear on the status line. Next, press the upper- or lowercase alphabetic character you've chosen to activate the double-width capability (in my example, D). Finally, move the cursor to the end of the heading, press F3 again, and press the key you've chosen to deactivate the double-width feature (d, in my example).

Your table can consist of 52 different macro definitions, each of which can be from one to 20 characters long.

CUSTOMIZING MACROS

Before running Define Macros, make a list of the alphabetic characters you want to represent the various functions your printer can handle. These will be your macro characters. Then run the Define Macros program.

First you'll be prompted to select a macro character. Enter any upper- or lowercase character from A to Z. (If you make a mistake and wish to cancel a macro definition, use the * key.) You'll then be asked how many characters will be represented by the macro character you've entered.

Next, enter the decimal value of each character in the string, starting with the first and continuing until all have been entered. Once you've done this, you'll have defined one macro. The prompt, "finished all definitions (y/n) ?" will then appear.

DOT COMMANDS

RUN Script Plus is a post-formatted

word processor. This means that the text is not formatted until it's printed, so your screen display will vary from your printout. To specify output format, RUN Script Plus has about 40 dot commands, so called because each command must be preceded by a dot (a period). These commands are embedded in the text to specify margin widths, define headers and footers, and so forth.

A dot command is executed when text is printed, and only text following the dot command is affected.

Four steps must be followed for dot commands to be interpreted correctly. First, the line immediately preceding a line of one or more dot commands must end with a return. Second, the line of dot commands must start in the first screen column. Third, multiple dot commands in a string must not be separated by spaces. Fourth, each string of dot commands must end with a carriage return.

The dot commands are:

.lj—Left-justifies printed text.

.rj—Right-justifies printed text.

.cn—Centers printed text between the left and right margins. Your text must begin on the next line below the .cn command.

.pl—Sets the page length. This command must be followed by a number from 1–240, to set the number of lines that will constitute a full page.

.pw—Sets the page width, defined as the maximum number of characters printed on one line.

.lm—Sets the left-margin width. The .lm directive must be followed by a number from 1–240.

.rm—Sets the right-margin width.

The `.rm` directive must also be followed by a number from 1 to 240.

.tm—Sets the top-margin depth. It must be followed by a number from 1 to 240.

.bm—Sets the bottom-margin depth and must be followed by a number from 1 to 240.

.hd—Defines a header to be printed at the top of every page. The command may be followed by up to 255 characters and must end with a carriage return. This means that the carriage-return character cannot be part of the header string and that no dot commands can follow the `.hd` directive, since they would be interpreted as part of the header string. For this reason, and to enhance readability, I recommend that the `.hd` and `.ft` (see below) dot commands appear on lines by themselves. The `#` character following the `.hd` (or `.ft`) has special significance. Entered just once at the beginning of a document, `.hd#` automatically prints the page number of each page.

The `.lj`, `.rj`, `.cn`, `.pw`, `.lm` and `.rm` dot commands have no effect on the header. The header string is printed exactly as defined, starting in the first column on the page and continuing until it has been printed in its entirety. If, for example, you wish to center a title, the title must be preceded by the correct number of spaces.

You may embed macro characters (see discussion above) in the header string if you wish. The header will be printed at the line position equal to the `.tm` setting plus 1. So, if you set the top margin to 6, the header will be printed on the seventh line.

.ft—Defines a footer to be printed at the bottom of every page. It works exactly like that of the header. The footer will be printed at the line position equal to the `.pl` setting minus the `.bm` setting minus 1.

.hs—Defines the number of lines to be left between the header and the main body of text. The command must be followed by a number.

.fs—Defines the number of lines to be left between the footer and the main body of text.

.ls—Sets the line spacing. You can print one or more blank lines between each line of text. For double-spacing (one blank line between lines of text), set `.ls` to 1.

.l+—Indents text from the current left-margin setting. For example, if the left margin is set with `.lm8` and you execute `.l+3`, text will be indented as though you'd set `.lm` at 11. To cancel an indent, use `.l+0` or `.l-0`.

.l-—"Outdents" text, such as a subheading, to the left of the left margin. For example, if the left margin setting is `.lm8` and `.l-3` is executed, text will begin printing at the sixth column, just as though `.lm` had been set at 5. Outdents are canceled with `.l-0` or `.l+0`.

.fp—Forces a new page. When this command is executed, no more text will be output to the current page. If a footer was defined, the footer and bottom margin will be printed, and then a new page will be started.

.fl—Links files to be printed. The command must be followed by a device number and a filename, separated by a comma. When the `.fl`

directive is executed, the specified file will be loaded from the specified device and begin printing. The permissible device numbers are 8 or 9 for disk. If two disk drives are used, one document could even slightly exceed 340,000 characters in length.

.p#—Sets the page number of the next page to be output.

.lf—Prints a line-feed character after every carriage return. Some non-Commodore printers require this. (.lf1 enables line feeds; .lf0 disables them.)

.cm—A handy dot command that lets you leave a comment for yourself that won't be printed.

.r+—Indents the right margin. This code must be followed by a number from 0 to 255. For example, if you set the right-margin width to 10 spaces with .rm10, and then execute .r+5, the effect is the same as if you had executed .rm15. A right indent is canceled by executing .r+0 or .r-0.

.r—Makes a right outdent. The command must be followed by a number from 0 to 255. For instance, if you've set the right margin to 10 with .rm10, and then execute .r-5, the effect is the same as if you had originally executed .rm5. A right outdent is canceled by executing .r+0 or .r-0.

.bj—Stands for "both justify" and prints the ensuing text with both the left and right margins justified. You cancel the .bj command by executing an .lj, .rj or .cn command.

.pr—Sends a sequence of up to 98 bytes to the printer using a specified secondary address. The secondary address must immediately

follow the command; then the bytes to be sent to the printer, separated by commas, must follow after the secondary address.

.ta—Stands for "text address" and must be followed by a number from 0 to 31. The number is the secondary address that will be used to print the ensuing text, and it lasts for only one printout. You can set the default (permanent) text secondary address with the F1,T command.

.ca—Defines a new, temporary secondary address. This is the secondary address that will be used to send macro strings to the printer and to print dot-graphics characters. Set the default (permanent) secondary address with the F1,C command.

.dc—Stands for "define character." This command takes three forms that enable you to redefine any character on the keyboard except @, to a total of 127 characters, then to undefine them. The total is enough to redefine the entire keyboard if you wish.

Example: .dca,66. To define a character, follow the .dc command with the character to be redefined, a comma and the new decimal value of the character. The Commodore ASCII decimal value for the letter "a" is 65. Here, the value has been changed to 66, which is the value for "b." Therefore, at printout time, all the a's in the document will print out as b's.

Example: .dca-. The minus sign following the character undefines that character. This example would undefine the character "a," removing it from the table of redefined characters.

Example: `.dc@`. This would erase the entire table of redefined characters, effectively undefining all of them.

.lr—Stands for "load redefined characters." This command is used only after you've performed three steps. First, you must define a number of characters with `.dc` commands. Second, you must print the current document; third, you must save the table of redefined characters to disk with the `F1,R` command. Then you can use `.lr` to load the saved characters during a later printout. This is useful when you have to redefine many characters—to access special characters on a printer, for example. The `.lr` command must be followed by a device number, a comma and a filename.

Example: `.lr8,filename`. This example would load the file of redefined characters named "filename" from device 8, the disk drive.

.lc—Stands for "load characters" and enables you to load a new character set or a new print set during printout. The `.lc` command must be followed by a number from 0 to 9 (0–6 for a C-64) that specifies the set to be loaded. A 0 loads a new set onto the screen; the numbers 1 through 9 load a new set into the printer. After the number comes a comma, followed by the device number from which the set is to be loaded, then another comma and the filename of the character or print set.

.el—Stands for "empty lines" and prints the specified number of carriage returns.

.st—Stands for "stop." This command works only with the printer or

the monitor, not the disk drive.

Example: `.st`. When not followed by any parameters, the `.st` command terminates Continuous mode during printout and starts Single Sheet mode after the current page is done. The command may be placed anywhere within the text area.

Example: `.st5`. When followed by a parameter, `.st` interrupts continuous output at a specified page number, anywhere from 1 to 65535, and enters Single Sheet mode. In this example, if continuous output has been selected with `F1,p` or `F1,o`, the first four pages will print in Continuous mode, then at page 5 the "next output (c/p/s) ?" prompt will appear.

.po—Stands for "page order" and must be followed by three numbers, separated by commas. This command enables you to print on both sides of a page or in a number of columns, as in a newsletter. Any number of columns can be printed across one page—2, 3, 4 or even more. The `.po` command works only when continuous output is selected.

Example: `.po1,2,1` and `.po2,2,1`. This combination of `.po` settings lets you print on both sides of the page, with text extending across the full width of the page.

The first setting is for the first pass. The first parameter in this setting sends the first page to the printer or disk; the second parameter specifies that every second page after that will also go to the printer or disk. For example, the odd-numbered pages might be printed and the even-numbered pages not—a phenomenon I call "page cycle." The last parameter in the first setting indicates how

many pages will be output before the page number is incremented. It should equal the number of columns across the page.

The second setting is for the second pass. Notice that the last two parameters are unchanged. The new first parameter, however, now sends the even-numbered pages to the printer. You print on the same paper as in the first pass, but now you use the back side. You can even define different headers or footers for the odd- and even-numbered pages—to place page numbers on opposite sides of the page, perhaps.

Example: `.po20,1,1.st21`. You can combine the `.po` and `.st` commands to print only one page out of a document. In this example, pages 1–19 will not be printed, but page 20 will. After page 20, the `.st` command will bring up the “next output?” prompt, at which point you can abort the print operation.

.ff—Stands for “form feed.” The `.ff` command must be followed by number 0 or 1. If you select 1, the bottom margin will be printed by sending the form-feed character to the printer. The Default mode, `.ff0`, prints the bottom margin with carriage returns.

.fc—Stands for “force conditionally” and must be followed by a number from 0 to 255. If less than the specified number of lines remain on a page when `.fc` is executed, no more text will be printed on that page. Instead, a force page will be executed, printing the footer (if any) and bottom margin immediately and resuming text on the next page.

.dg—Stands for “define graphics character.” This command lets you change any character, in any print set or the screen character set, without using a character-set editor.

The command must be followed by ten byte numbers, separated by commas. The first byte number specifies the set in which the character to be changed is situated. As always, 0 specifies the screen character set, 1–9 specify a print character set.

The second byte is the screen-code value of the character to be changed. The last eight bytes define the character itself.

.gc—Stands for “graphics characters.” The `.gc` command is followed by two parameters, separated by a comma. The parameter settings are saved within RUN Script Plus, and are permanent until the computer is turned off, or until you execute another `.gc` command.

The first parameter is a number, 0 or 1, that specifies the orientation of dot-graphics characters printed when you select either a print set or Special Graphics mode. Only one setting is correct for any one printer. If you select the wrong setting, all characters printed in Graphics mode will be printed upside down. The only way to determine the correct setting for your printer is by experimentation.

The second parameter sets the width of Graphics Mode characters in dots. Standard Commodore characters are eight dots wide. So, if you set this parameter to 6, only the first six dots of each character will be printed. If you set this parameter to

10, ten dots will be printed—eight dots of character and two dots of space.

To make use of the full eight-dot width of a Commodore character and print 80 characters per line, you need a printer with a dot density of $8 \times 80 = 640$ dots per line. Use a dot width of eight if you have such a printer.

In general, use the following formula to calculate the correct dot width: dot width = dots per line ÷ characters per line. If the result is a dot width of less than eight, you'll need a print set that uses only that many columns of a Commodore character.

.gb—Stands for "graphics begin" and defines the string of bytes that is sent to the printer to activate Graphics mode. The string is sent before each character is printed in Dot Graphics mode.

The MPS-803 and its compatibles require just one character, `CHR$(8)`, to activate graphics mode. This character will put the printer into Graphics mode until it receives a byte value of less than 128, at which point it will return to normal Text mode. Most other printers handle graphics differently, however, and require that more than one character be sent.

.ge—Stands for "graphics end." The string of bytes following the `.ge` command will be sent to the printer after each dot-graphics character. Some printers require this command to return to Text mode.

PRINTERS

RUN Script Plus expects a Com-

modore printer connected to the serial bus. A combination of a non-Commodore printer and an interface that emulates a Commodore printer will also work.

If your system includes neither of these, you can still use RUN Script Plus by putting the printer interface into Lock mode and using the RUN Script Plus `F1,t` function to output true ASCII. Another method is to use the Define Macros program (see previous discussion) to create a customized set of macros that RUN Script Plus can use to control your printer.

Note that you cannot use an RS-232 printer with RUN Script Plus. If you have such a printer, you must output your text to disk and then use a Basic program to print it.

DAISYWHEEL PRINTERS

RUN Script Plus can do underlining and double-strike characters with daisywheel printers that recognize the back-space character. Type `F3`, then the `(` character to start underlining; type `F3`, followed by `)` to end underlining. Type `F3`, then `[` to start double-strike; type `F3`, followed by `]` to end double-strike. (See also the `.bs` command.)

To change the print wheel, type `F3`, then `*`. When RUN Script Plus encounters the reversed `*` character, it will stop until you press a key.

GRAPHICS MODE

Most dot matrix printers can print dot-addressed graphics and ordinary text on the same line, and RUN Script Plus takes advantage of this feature. When you put RUN Script

Plus into a graphics mode, instead of outputting normal text, it uses the printer's dot-graphics capability to print each character. This allows printers such as an MPS-803 to print italics, boldface or underlining and print foreign character sets.

There are two ways to create graphics with RUN Script Plus. One is through a "print set," which is half of a normal character set. A character set contains 256 characters, 128 non-reversed and 128 reversed, so a print set contains 128 characters. How does it work?

You select a print set with the F3 key, the same key that selects macro characters. After you press F3, choose a numeral from 0 to 9. The numeral will appear at the cursor position in the text, in reverse field.

Pressing 0 selects normal Text mode, which is the default; pressing 1-9 enables the Graphics mode. Within that mode, 1 specifies the first print set, 2 specifies the second set, and so on, up to the maximum number of sets your computer can have (six for the C-64, nine for the C-128).

For example, say you chose print set 3 and the next character to be printed is an a. The letter a is the second character in the standard Commodore character set (see the *C-64 Programmer's Reference Guide*, page 376, for a listing of the character set), so RUN Script Plus will go to set 3, take the second character of that set, and print it using dot-addressed graphics.

Only six print sets are available in the C-64 version of RUN Script Plus, but you can still press the 7, 8 or 9 key after the F3 key. If you press 7

or 8, RUN Script Plus will use the non-reversed characters of the character set as a print set; if you press 9, it'll print the reversed characters.

Print sets and the character set are loaded into memory by the boot program, which also loads the RUN Script Plus machine language program and initializes the RUN Script Plus system.

You also can print graphics with the Special Graphics mode. It's activated by the F1 key, then g, then answering y at the prompt "enable special graphics (y/n) ?".

When the Special Graphics mode is functioning, the business-graphics characters of the character set, which are accessed by simultaneously pressing the Commodore logo key and any other key, will be printed using the printer's Dot Graphics mode. This occurs only when normal Text mode is selected. All other characters will be printed as normal text characters.

NEW RUN SCRIPT COMMANDS

There are three new key-press commands (preceded by F1) for the main RUN Script Plus program: 1, \$ and Q (which accesses the spelling checker). (The \$ command is not really new, but it now functions differently.) In addition to the new commands, the <, > and @ commands have been modified to work with Commodore's RAM disk software. There is also one new dot command, .lb, which activates the label printer.

1—Press F1, then 1. This command saves your text twice, first to device eight for permanent storage, and then to device nine, which is

normally the device number of a RAM expander when used as a RAM disk. After your text is saved to the RAM disk, it can be loaded in about one second or less. (Note: At this point, the RAM disk software for Commodore's 1700 and 1750 RAM expander cartridges for the C-128 has not been formally released, so we cannot guarantee that the RAM disk capability will work properly for the C-128. This discussion of the RAM disk is therefore at present applicable only to the C-64.)

To use RUN Script Plus with Commodore's 1764 RAM expander employed as a RAM disk, you first run the RAM disk initialization program, pressing return at each prompt. Then you load and run RUN Script.

\$—Press F1, then **\$** to load a directory from disk into memory. The directory is displayed as it loads in. Press the run-stop key to terminate the load when the desired filename appears. Then position the cursor over the filename. Next, press the appropriate key from the list below to perform the action you want:

- I**—Load a text file.
- r**—Read in a sequential or user file.
- >**—Issue a disk command.
- <**—Read the disk error channel.
- m**—Load a macro table.
- 0**—Load a character set.
- 1 to 6**—Load a print set.

After you enter a command letter, a prompt appears. Pressing the F3 key reads the selected filename from the screen directly into the input line.

However, this command has a few idiosyncrasies. First of all, the directory is loaded into text memory. If

an insufficient amount of memory remains, a Text-Area-Full message appears, and only a portion of the directory is loaded.

Always exit the Directory mode by pressing the Commodore and control keys simultaneously, thus erasing the directory. Pressing the restore key leaves the directory embedded in your document.

Q—The command F1,Q accesses the spelling checker. (See the section on the spelling checker, below.)

.lb—The dot command **.lb** activates the label printer. RUN Script Plus can now generate mailing labels. You must enter the labels into the RUN Script Plus text area, pressing return at the end of each line and two returns at the end of the last line of each label.

After the program encounters the **.lb** dot command in the text, it activates a special printing routine that only recognizes three dot commands: **.pl**, **.pw** and **.lb**. The **.pw** command is used in the normal way.

The **.lb** command requires a variable number of parameters, but it is always followed by at least two numbers. The first number, from 1 to 9, specifies the number of labels to be printed across the page. The next number specifies the starting column position of the first label. The columns are numbered starting with zero.

THE SPELLING CHECKER

The most significant addition to RUN Script is the spelling checker. It works by comparing each word in a document with a list of words stored in a dictionary. Any word not

located in the dictionary is marked. Later, you can edit these words if misspelled, replace them, add them to the dictionary or ignore them.

A short dictionary on this disk contains about 3000 words. You can store a dictionary of about 35,000 words on a 1541 drive, and a maximum of about 75,000 and 172,000, respectively, on the 1571 and 1581 drives.

"Personalized" dictionaries that contain only those words most commonly used are recommended.

With 10,000 words or less in the RUN Script Plus dictionary, spelling checks with a 1541 drive take about one to two minutes to check a 100 block document. If you're using a C-64 with a RAM expander, you can copy a personalized dictionary to the RAM disk for incredibly rapid spelling checks.

Another technique that increases the efficiency of spell-checking is to limit the process to checking only those words having lengths equal to or exceeding a user-defined limit.

RUN Script Plus's minimum length default value is five letters, although the length is adjustable from three to nine characters.

The spell checker recognizes as words only those groups that are formed solely of the letters a-z and A-Z. Consequently, contractions cannot be placed in the dictionary; they would be interpreted as two words.

Two methods exist for activating the spell checker. One way is via the F1,Q command from within RUN Script Plus. When this command is used, any text is preserved, and it's

possible to exit from the spell checker back to RUN Script Plus by using the F1,x command. Another way is by running the spell checker as a stand-alone program.

COMMANDS AND OPERATION

Nine commands are available within the spelling checker, all of them activated by pressing the F1 command key, and then the appropriate letter, just as in RUN Script Plus. You cannot edit text while using the spell checker; instead, use RUN Script Plus for that purpose. Some of the nine commands are identical to those in RUN Script Plus, with one area of procedural difference: All commands that access the disk drive require you to enter a device number to make working with two disk drives easier. The nine spell checker commands are listed after the following description of the spelling checker operation.

Q—Press F1 and then Q to activate the spelling checker. With the C-64, the spelling checker program prompts you for the device number for the spell checker, then to insert the correct disk and press a key. In a few seconds, you'll be able to proceed with the spelling check.

Press F1,Q again and you'll be prompted to enter the device number of the drive containing the dictionary. After you do this, the spelling check proceeds with its work. Words in your document that have a length equal to or greater than the specified minimum length (see command F1,W, below) will be compared with words in the dictionary. Any words not in the dictionary are marked.

After the check is completed, each word that was marked becomes highlighted on the screen, and a prompt appears presenting the available options: 1—edit word; 2—add word; 3—skip word.

Pressing the 2 key saves the word for later addition to the dictionary. Pressing the 3 key skips the word, so it is neither edited nor added to the dictionary.

Press the 1 key when you encounter a misspelled word. An "edit ?" prompt is displayed. If you then press the F1 function key, the word appears on the input line, where you can edit it. After you've corrected the spelling, you press the return key to terminate the edit process, and the following prompt then appears: 1—prompt; 2—replace all; 3—skip word.

If the 1 key is pressed, a "replace (y/n) ?" prompt is displayed before each occurrence of the word. This allows you to replace words on a case-by-case basis. After all occurrences of a word have been dealt with, the next marked word will be displayed, and the process repeats.

After you've either edited, skipped or added all the highlighted words, the time comes to save to disk all the words you designated to be added. First, prompts appear requesting a device number and a filename; then the list of words is saved. The list should be saved on the dictionary disk. However, the words do not actually become part of the dictionary until the DICT MAINT program (see below) has been used on them.

THE NINE COMMANDS

The nine commands, all preceded

by the F1 command key, that you can use when within the spelling checker are as follows:

I—Load a text file.

S—Save a text file.

@—Scratch the old text file, then save a new copy to disk.

<—Read the disk error channel.

>—Issue a disk command.

\$—Print out a disk directory listing.

X—Exit to Basic or RUN Script Plus.

E—Counts the number of *unique* words in the document, with the exception of those below the minimum specified length (see the F1,W command, below); if you then press F1 again, you get the total number of *all* occurrences of *all* the words in the document.

W—Sets the minimum length for the words that will be checked.

DICTIONARY MAINTENANCE

The dictionary maintenance program, called RS DICT MAINT, is used to add new words to the dictionary. The list of words to be added can be the file created by the spelling checker, or you can create it with RUN Script Plus by typing words in, then saving them to disk. If you use this method, the RUN Script Plus file must be loaded into DICT MAINT, then sorted with the F1,O command (see F1,O, below).

When you're typing dictionary words into RUN Script Plus, you must ensure that each word is separated from other words by some non-alphabetic character, such as a space or carriage return.

Twelve commands in the DICT

MAINT program are activated by first pressing the F1 command key, then the appropriate key listed below.

I—Load a text file.

s—Save a text file.

a—Append a text file to the end of a text file already existing in memory. Note that this operation differs from the way the F1,a command operates in RUN Script, where the file would be loaded beginning at the cursor position. This command permits several lists of added words to be loaded and then merged into one large list, using the F1,O command (see F1,O and F1,D, below).

\$—List the disk directory.

>—Issue a disk command.

<—Read the disk error channel.

x—Exit to Basic.

E—Count the number of unique words and the total number of all words in the document currently residing in memory.

O—Sort any words in memory into alphabetical order, then save the entire sorted list to disk. It is necessary to use this command after combining several lists of words using the F1,a command because the F1,D command, which merges new words into the dictionary, requires a sorted list of words on disk.

Also, use this command on lists of words previously created with RUN Script Plus, perhaps copied from a dictionary. Doing this ensures both that the list is sorted and that it's stored in a format compatible with the F1,D command.

U—Unpack a dictionary file into text. RUN Script Plus dictionary files are stored in a packed format that

allows 2000 to 3000 words to be stored in only 8000 bytes, which is the maximum for one dictionary file.

The purpose of unpacking a dictionary file is that it allows text to be edited with RUN Script Plus. Words can be freely deleted or added as you wish. Just make sure that improper words are not added to a particular file. For example, words beginning with C are divided into two files, "RS.C" and "RS.CM". You must make sure that no words starting with CM, CO, CP, etc., are placed into the "RS.C" file. If "RS.A" is being edited, be certain that no words that begin with other letters are added to that file. If this rule is not followed, the spelling checker will not function properly.

Use the following procedure if you wish to edit a dictionary file:

(1) Load and run the DICT MAINT program. Press F1, then U.

(2) Enter the filename corresponding to the desired dictionary file, from "RS.A" up through "RS.Z."

(3) After the file is unpacked, save the unpacked file with either the F1,s or F1,@ commands.

(4) Edit the file with RUN Script Plus and re-save it.

(5) Load and run the DICT MAINT program.

(6) Load in the edited file.

(7) Press F1,O to sort the words and save them to disk.

(8) Load in the file saved with the F1,O command. At this point you can, if you wish, press F1,E to see how many words are in this file.

(9) Use the F1,P command to re-pack the file into dictionary format.

Make sure the file is assigned the correct filename.

P—Packs a list of words into dictionary format. The words must be in alphabetical order, and each word must be followed by a carriage return character (see F1,O).

D—Merges a list of words into the dictionary. The list must already exist in the correct format; e.g., the list must be created either by the spelling checker or by the F1,O command. Lists of words created with RUN Script Plus require processing through the F1,O command before they are acceptable. The word list and the dictionary files (if any—see Personalized Dictionary, below) must share the same disk.

First, a "device number for dictionary?" prompt appears. Enter the device number containing the dictionary files. This can be a conventional disk drive, a 1581 3½-inch drive or a RAM disk. Next an "added words filename?" prompt appears. Enter the filename of the list of words you want merged into the dictionary. Finally, a "device number for new dictionary?" prompt is displayed. Enter the device number on which to store the new dictionary files.

Be forewarned that dictionary files larger than 8000 bytes can crash the spelling checker.

PERSONALIZED DICTIONARY

For maximum speed, it's best to create a personalized dictionary, which contains only words that you

commonly use. Start by loading and running the spell checker. It doesn't matter that no dictionary exists. Just use any blank, formatted disk as a dictionary disk. Next, load in four or five of your old RUN Script text files and use the F1,Q command on them. If you do not have any old text files, create one with RUN Script Plus by typing in common words from a dictionary (the book type).

After you've created a few files of added words, exit the spell checker and run the DICT MAINT program.

Use the F1,a command to load all your files of added words into memory. Keep in mind that the DICT MAINT text area holds only about 35,000 characters, or between 4000 and 5000 total words. Also DICT MAINT can only handle about 4000 unique words at one time. Use the F1,E command to be certain these limits are not exceeded.

After you place a large file in memory, execute the F1,O command to write the unique words out to disk. Next, execute the F1,D command. Since no dictionary file exists as yet, the initial use of F1,D creates one and gets you started.

OVERLAYS

The figure on the inside front cover shows a set of RUN Script function key overlays for the C-64 and the C-128. Cut out the appropriate one and place it over the function keys on your computer to make RUN Script Plus easier to use. ■

RUN File 1.0

By David Darus



C-64; C-128; printer (optional)

RUN File 1.0 is a database program for the C-64 and C-128 that's designed to do all the management tasks involved in keeping records such as mailing lists and inventories. It maintains the records in an indexed order and provides utilities to manipulate the data into sets of information.

The program requires a C-64 or C-128, a monitor and a 1541, 1571 or 1581 disk drive. A printer is optional.

To use RUN File 1.0 on a C-64, enter the command LOAD "RUN FILE.C64",8 and the Run command. To use the program on a C-128, just enter RUN "RUNFILE.C128". The C-128 version automatically adjusts for 40- or 80-column operation.

After some set-up time, the Main menu screen appears. The work area is in the top 22 lines of the screen, with program and database identification next, followed by the Main menu command line and a message line.

The command line offers six options: Close, Edit, New, Open, Utils and eXit. To select a command, ei-

ther press the capitalized letter in the command, such as N for New or X for eXit, or use the right-and-left cursor key to highlight the command, then press return.

There are some restrictions on when commands can be executed. You can't close or edit a database or execute the Print, Seq and Usr commands unless you've opened the database first. Also, you can't open more than one database at a time, and you can't create a new database or exit the program if a database is open.

NEW

The New command is for creating a new database form. You'll be prompted to enter a filename that's anywhere from one to 14 characters long for the database, and the program will automatically give the filename a .f extension. (Don't include the extension when you type in the filename.)

In the first 22 lines of the screen, you can define and fill up to 30 fields, each having a maximum length equal to the screen width minus 1—that is, either 39 or 79 characters. The total length of all fields in the record can be up to 254 characters long, not including field

names. As you type in text, use the delete key to correct mistakes.

To define a field, type its name (such as LAST NAME or STREET). To define the length of the field, press control/F to mark its start, press the space bar as many times as you want and press control/F again to mark the end. As you do this, you'll notice a field-length counter and a total-record-length counter at the bottom of the screen. You can use the delete key to shorten a field as you're defining it, but you can't use it to erase the field name. To do that, you must press control/D to blank the line out. The total record length will be adjusted accordingly.

Any time you press the enter key on a line, the part of the line that isn't highlighted becomes the new content of that line and the highlighted region becomes blank. For this reason, use the cursor keys to bypass lines you've already edited and wish to keep as they are.

When you're satisfied with the layout of the line, press return for the computer to accept it. If you press the cursor up-and-down key without pressing return, the line won't be accepted.

When you're satisfied with the entire form on the screen, press the run-stop key in 64 mode or the escape key in 128 mode. The current field will be highlighted, and the following list of field types will appear at the bottom of the screen: Alpha Key Num Special. You'll be prompted to choose a type for each of the fields in the form by pressing

the capitalized letter of your choice.

You can select the Key attribute only once, and it should be assigned to the field that distinguishes records from each other, such as record number, employee ID number or inventory number. If you don't specify a key field, the program will automatically assign the first field in the database to that purpose. Other fields can be assigned one or more of the other types—Alpha, Num and Special. You can cancel a field-type selection for the current field by pressing the capitalized letter again.

Alpha fields can contain upper- and lowercase letters A-Z and spaces. Num fields can contain () * + , - . / 0 1 2 3 4 5 6 7 8 9 and spaces. Special fields can contain ! # \$ % & ' () * + , - . / : ; < > = ? @ [] | _ £ and spaces. Once you've set the types for the current field, press run-stop in 64 mode or escape in 128 mode to make the computer accept them and go on to the next field. When you've specified a type for every field, the form will be saved to disk and you'll return to the Main menu.

OPEN

The Open command is used to access a particular database and to read its form and index into memory. When you choose Open at the Main menu, you'll be prompted for the name of the database you want. (Again, don't include the extension to the filename.) The first time you open a database, you must estimate the number of records you'll want to put in the file, so disk space can be

allocated. This estimate by no means limits the number of records you can enter, as long as you don't exceed the maximum of 1000 records in 64 mode or 2000 in 128 mode.

After you've entered your estimate, an index file (with a .i extension on the filename) will be created, along with a relative data file (with a .d extension).

EDIT

The Edit command lets you access, browse through and update records. When you enter Edit mode, the cursor appears on the first field in the form. Use the up-and-down cursor key (keys, in 128 mode) to move from field to field, without disturbing the contents of the fields. After you've typed in or altered the content of a field, press return.

Edit mode provides a number of function-key commands, including Find, Write, First, Last, Next, Previous, New and Print.

Find (F1) is a search function. When you press F1, the program prompts you to select (via the cursor keys) the field you want to search by. Press return to highlight the field you want, then press the run-stop key in 64 mode or the escape key in 128 mode. Next, you'll be prompted for the search string. Enter only as much of the string as is needed to make it unique, and keep in mind that the case of the letters is significant; then press return. If the search field is an index key field, the program goes directly to the matching record, if present, or to the next highest record. If the field you select

isn't an index key field, the disk is searched for a match. When one is found, press return to find the next matching record, or press run-stop (C-64) or escape (C-128) to abort the search. While in this mode, you can press F8 to print the screen.

Write (F2) stores the record using the index key that's in the key field, along with all the other field data. In Write mode, you can change an existing field by typing over it, pressing return and then F2. Depending on the content of the index key field at that time, a new record will be created or an existing one overwritten (if it has the same key field contents). After editing the last field you want to change, press F2 instead of return.

First (F3) displays the first record in the database, according to the index key value.

Last (F4) displays the last record in the database, according to the index key value.

Next (F5) displays the next record in the database with an index key field greater than the current one.

Prev (F6) displays the previous record in the database with an index key field less than the current one.

New (F7) blanks out all data in the current form, leaving an empty form for entering new data.

Print (F8) prints the first 22 lines of the screen on the device 4 printer.

To exit Edit mode, press run-stop in 64 mode or escape in 128 mode.

UTILS

The Main menu Utils command brings up a submenu of the follow-

ing commands: *Dir*, *Dos*, *Drive#nn*, *Print*, *Seq* and *Usr*. To select one of these commands, use the cursor keys to highlight it and press return. (Remember, the *Print*, *Seq* and *Usr* commands can be executed only if a database is currently open.) You can press run-stop in 64 mode or escape in 128 mode to return to the Main menu from *Utils*.

To select fields to be output by the *Print*, *Seq* or *Usr* command, move the cursor to each field you want and press return. When you move the cursor from that field, the field will be highlighted. You can't cancel a field selection, so be careful in making your choices. You can select the fields in any order—the key field has no effect here—and the order you choose will determine how the output records are built.

The *Print*, *Seq* and *Usr* commands include a *Sort* option. If you choose this option, the order in which you select fields will determine the sorted order of the records that are output.

Dir displays the directory of the disk in the drive. In C-64 mode, pressing the space bar once pauses the directory scrolling and pressing it a second time reactivates the scrolling. In C-128 mode, use the no-scroll key to pause the display. When you're done with the directory, press return to go back to the form.

Dos lets you issue DOS (disk) commands to the current drive, and these use Basic 2.0 syntax, as described in your disk drive manual.

The *Drive#nn* command, where *nn* is the current device number of the disk drive, changes the device

number to any number from 8 to 11. Just enter the number you want at the prompt and press return. If you enter a number that's out of range, the number won't change. The current device number is displayed on the *Utils* command line. The *Drive* command enables you to use multiple drives for your files.

Print produces hard copies of your data in report or label format. Report format prints all the data in a horizontal line, while label format prints one line at a time. The *Print* command lets you select fields in the order you want them printed, if your printer is device 4. Also, you can sort the records in the database before printing. If you're planning to use the *Print* option for mailing labels or as a report generator, keep the final printed output in mind when you're designing the database form.

Seq creates a sequential file of selected fields, in the order you select them and sorts them if you wish. The format of a *Seq* file is one field of data, followed by a carriage return. There's an extra carriage return at the end of each record in the file. The *Seq* files can be read directly into many word processors.

Usr creates a user file of selected fields, in the order you select them, and sorts them if you wish. These files can be used in your programs.

CLOSE

The *Close* command in the Main menu closes a database, updates the index on the disk and clears the form from the screen. This command is especially important because of

the index update. As a safeguard, you can't exit RUN File 1.0 without closing the current database. However, this safeguard doesn't work during a power failure. If the power goes out, the index won't be updated and all the records you've added to the database since the last closing will be unaccounted for in the index. Close and open your database periodically to update the index when you're doing a lot of data entry at one time.

EXIT

The eXit command is used to leave RUN File 1.0.

IMPROVING PERFORMANCE

Performance of the database can be improved by proper design of the file form. Making the key field small and placing it first in the form, as well as making the entire record small, improves speed. Also, it's good database practice to design your form on paper first, resolving any problems before defining it in the program. This will save you a lot of work and frustration.

The figure on the inside back cover is a set of RUN File 1.0 function key overlays for you to cut out and use with your C-64 or C-128 computer. ■

RUN Calc 128

By Mike Konshak

RUN It Right

C-128

RUN Calc 128 is a spreadsheet for the C-128 for managing home budgets and keeping track of expenses.

Spreadsheets such as RUN Calc 128 organize data in cells, which are located at the intersections of rows and columns. Each cell is identified by the row and the column it's in. The concept of a cell is the primary difference between a spreadsheet and other types of programs.

A cell can contain one of two types of data: a label or a value. Labels are alphanumeric text data such as names, addresses, quantities, ages, dates, and so forth. A value can be a number entered into a cell and used in obtaining a result that appears elsewhere on the spreadsheet. A value can also be a formula or equation assigned to a cell, the result of which appears in that cell after a calculation has been performed.

Formulas are hidden and don't appear on the spreadsheet itself, although they can be viewed and edited when you position the cursor at that particular cell. RUN Calc 128

recognizes a number or a formula as a value if it's preceded by a plus (+), minus (-) or at symbol (@). Otherwise it's treated as a label. Labels are normally left-justified within a cell, whereas values default to right justification with two decimal places, which brings us to another nice feature of RUN Calc 128.

The cells within each column of a spreadsheet can be formatted specifically for the type of data that will appear within it. RUN Calc 128 is no exception and permits several types of justification and ways to present numerical data. Also, the width of each column can be set according to the length (in characters) of the data that will reside in the cells in that column. Because a spreadsheet can combine text like a word processor, repeat data like a database and make calculations like an adding machine, it's sufficiently versatile to create many types of forms (to fill in the blanks later) and reports.

RUN Calc 128 behaves like a commercial spreadsheet. You can write the formulas exactly as you would write a Basic expression, and you can enter the special functions, like @SUM(A1..Z250). In an effort to appease fans of Datafile, RUN's database program, I've provided for

importing up to 250 Datafile record files.

When you first run RUN Calc 128, you'll notice that rows are numbered 1-250 and columns are labeled A-Z. Cells are designated by column, then row, such as A1, B26 and Z250. There are 6500 cells available, although you may run out of memory before you use that many. Column widths default to ten characters, but may be changed.

Whenever an error occurs or you press an inappropriate key, the computer beeps (if you have the monitor's volume turned up). Also, any time the computer is working, a Wait message flashes in the upper-right corner of the screen. A nonflashing Ready message means the computer is ready to accept input from the keyboard. You'll also encounter other status messages (i.e., Disk, Print, Error, Calc1, Calc2, Edit, Input, Entry) as you perform various operations.

Additional status information as to the cell content, cell position and data type within the cell are shown in the upper-left corner of the screen.

GETTING OUT OF TROUBLE

Should you forget the function of a key or the syntax of a function, press the help key in the top row of keys. A full screen of explanations will be displayed to assist you. Press the escape key (or any other key) to exit back to your worksheet.

Pressing escape in the Worksheet mode displays the Main menu at the top of the screen. Once a menu is present, escape exits back to a pre-

vious menu or, ultimately, back to the worksheet. You can also obtain the Main menu by pressing the slash (/) key. Escape will also exit safely from disk drive routines and generally keep you out of trouble.

MOVING THE CURSOR

The cursor keys are used to move to a particular cell. The up-and-down cursor key positions to adjacent rows and the left-and-right cursor key moves to adjacent columns. This applies to both sets of cursor keys on the keyboard. Once you begin entering or editing the data within a cell, only the left-and-right cursor keys are active.

You'll find that moving horizontally or vertically through your worksheet becomes time-consuming if you move cell by cell. Pressing the control key and one of the four upper cursor keys will move the cursor to the adjacent screen in the direction noted by the arrows on the keys. Control plus up or down jumps 20 rows at a time. Control plus left or right displays the next group of columns. The actual number of columns that will fit on a screen depends on the width of the columns.

Pressing the home key once moves the cursor to the upper-left cell shown on the screen. Pressing home again moves the cursor to the beginning of your worksheet (cell A1).

ENTERING DATA

To begin entering or replacing data in a cell, move to the desired cell with the cursor keys and then

begin typing in your data. As I mentioned earlier, data is classified as a label or a value. If the data begins with a plus, minus, or at symbol, then the program interprets the data as a value. You'll notice that the characters you're typing in as values are automatically converted to uppercase.

After you press return to make the computer accept the data you've entered, it identifies all cells referred to by the formula and converts everything else to lowercase. The result of the value or formula is printed within the cell, normally right-justified with two decimal places. The contents of the cell, consisting of the data and the result, are also displayed at the top of the screen.

Immediately after the cell identification, the result appears, followed by a left arrow. After the left arrow, the data you entered for the cell is displayed. It may consist of a number value or a formula or function. A value entered as +25 appears on the status line as A1: 25— +25. A formula entered as +25*100/2 appears as A1: 1250— +25*100/2. The left arrow points from the formula to the result.

Labels are simpler and don't involve a result—or rather, the data you type in immediately becomes the result. Labels are normally printed left-justified within a cell, and they may consist of any character typed from the keyboard, including punctuation and Commodore graphics characters, but excluding the plus, minus, at and back-arrow characters, which denote values and for-

mulas. Only the graphics characters generated with the Commodore key are available, since the shift key produces uppercase letters.

When typing in labels, you don't have to keep track of the width of the columns, since RUN Calc 128 automatically overlaps text data into the columns to the right, assuming they're empty. If the column to the right already contains data, only the number of characters that will fit in the current cell are printed. The entire text string appears at the top of the page, even if it's too long to fit in the cell.

I'll explain how to expand the width of a cell a little later. If you do expand its width, as many of the original characters as will fit will automatically be printed in the cell. Because of the overlapping feature, you'll be able to write small paragraphs in your worksheet. The maximum number of characters you can enter inside or outside a cell is 74. Formulas in other cells that refer to cells containing numeric labels will perform correctly, even when the referenced cell data hasn't been identified as a value. Alphabetical characters possess a value of zero in these instances.

EDITING, ERASING AND EXITING A CELL

Once the cursor is positioned in a cell, pressing the return or enter key will let you edit the data within the cell. After you make your corrections to the data, return or enter exits the cell. If the cell contains a label, the label will be printed within the cell.

If it contains a value or a formula, a recalculation will occur and the result will be printed within the cell.

You can erase the data within a cell quickly by moving to the cell, then pressing return, the space bar and return once more. When you're entering data at menu prompts, you must always press return to accept the input.

RECALCULATION

Any time you exit a cell containing a value by pressing return, the formula within that cell is recalculated and the result displayed. If you've extensively altered your worksheet and want to recalculate it entirely, press the equals (=) key and go make a cup of coffee while the program analyzes and recalculates each of the cells within the current range. Two passes are required to ensure accuracy.

The status line displays Calc1 on the first pass and Calc2 on the second. Obviously, the larger the spreadsheet, the more time the recalculation will require. In most instances, it's faster to just recalculate individual cells by moving to the cells and pressing return twice.

MEMORY

Pressing the English pound (£) key displays the number of bytes free in the memory bank. When you're creating a large worksheet, periodically check the available memory. An Out of Memory error is the only one that can't be recovered, since it can't be trapped.

FORMULAS AND FUNCTIONS

Once again, RUN Calc 128 recognizes values and formulas by the presence of a plus or minus sign as the first character of an expression. A number, such as +100, is the most simple, producing a result of 100. A formula is more complex and might resemble $+E2 - F4 * (\sin(2 * 3.14159))^{12}$.

Most of the functions and operators that can be used in writing a Basic program can also be used in a formula within a cell. Because RUN Calc 128 uses the same Kernal routines that Basic programs use, a good rule-of-thumb is that if it causes a syntax error in Basic, it'll cause a syntax error in RUN Calc 128, so keep your *C-128 Programmer's Reference Guide* handy.

Basic expressions and formulas perform operations on regular variables, such as X, a% and RR, and on subscripted variables, such as a(1) and r%(2,2). A typical formula written in Basic might appear as $X = A + B$. In a spreadsheet, variables are substituted by cell coordinates.

In $X = A + B$, X is the cell where the cursor is currently positioned, and the variables A and B represent values located in other cells. If the cursor is positioned at cell A4 and you wish to add the contents of cells A1 and A2, A4 is shown to the left of your prompt, along with an equals sign. You type in $+A1 + A2$, and when you press return, the two cells are added and the result is printed within cell A4. Also, the cell coordinates, the result, a left arrow and the formula appear in the upper-left cor-

ner of the screen. Two rows below this status line, the program also informs you that this is a value.

OPERATORS

The Basic operators listed in Table 1 can be used in RUN Calc 128 formulas. They include arithmetic operators, as in $X = A + B$, and logical operators, as in $X = A < B$. In the latter example, if A is less than B, the expression is true and X is assigned a value of -1. If A is greater than or equal to B, the expression is false and X is given a value of 0. You can convert your result to a positive number by using the absolute function, such as $ABS(-1)$, which equals 1.

BASIC FUNCTIONS

Functions perform mathematical calculations on an argument. The argument is contained within parentheses and may be any number or cell coordinates. Refer to your *C-128 Programmer's Guide* for an expanded explanation of the functions. The functions in Table 1 are known to work properly in RUN Calc 128 formulas.

As you type in an equation, the characters appear in uppercase, but after you press return, only the cell coordinates remain capitalized. The functions in the table are always converted to lowercase letters, so the Kernal can understand them. Functions not allowed by RUN Calc 128 are obvious, because the function letters remain in uppercase.

The trigonometric functions, $ATN()$, $COS()$, $SIN()$ and $TAN()$, always return an angle in radians, not de-

grees. To convert a result in cell A1 from radians to degrees, use the expression $+A1*180/3.14159$. The parentheses are used exactly as in Basic program expressions.

@ FUNCTIONS

The @ functions simplify mathematical operations on a range, or block, of adjoining cells. For instance, rather than typing $+A1 + A2 + A3 + A4 + A5$, it's simpler to type $@SUM(A1..A5)$. The @ symbol must be the first character in the formula, preceding the three-letter function identifier. Cell coordinates must be enclosed within parentheses and separated by two periods (no spaces allowed). The available @ functions are also listed in Table 1.

The first cell set of coordinates is for the From cell, and the second is for the To cell. The @ functions can refer to some or all of the cells in one row or one column, or a block of cells consisting of multiple rows and columns. When you're defining a block of cells, the From, or first, cell must be in the upper-left corner of the block and the To, or second, cell must be the lower-right corner of the block.

To sum the values in all 250 rows and 26 columns of a worksheet, the function would be written as $@SUM(A1..Z250)$. An @ function can't coexist in the same cell with a conventional formula. This syntax is important for the proper operation of RUN Calc 128, and it will also be used later for determining which portion of your worksheet will be saved, printed or erased.

SAMPLE WORKSHEET

The sample worksheet in Figure 1 is typical of RUN Calc 128 applications. The income and expenses of the family are totaled and compared to produce discretionary income. Savings and retirement expenses are self-calculating, based on amount of income. Once the worksheet has been entered, it's a simple matter to see how much money is available to play with as income and expenses vary.

The sample worksheet adds the

Family Home Budget	
Dad's Income	2000.00
Mom's Income	1550.00
Total Monthly Income	3550.00
Mortgage	800.00
2nd Mortgage	200.00
Auto Loans	600.00
Insurance	80.00
Fuel	120.00
Utilities	150.00
Phone	35.00
Food	400.00
Clothes	250.00
Savings	355.00
Retirement	355.00
Total Expenses	3345.00
Discretionary Income	205.00

Figure 1. Sample RUN Calc 128 worksheet.

two family incomes together, producing a result of 3550. Savings and retirement expenses are calculated by multiplying the total income by .1, producing 355 for each item. The expenses are totaled, producing a result of 3345, which is subtracted from the total income to yield \$205 worth of "play" money for the month.

Move to the cell identified and type in the data as follows, pressing return after each entry. If you need to edit a cell, move to it, press return and make your changes. If you just start typing, you'll erase the previous data.

C1: Family Home Budget

A2: !! Press shift/* 50 times for a horizontal line.

A3: Dad's Income

C3: + 2000

A4: Mom's Income

C4: + 1550

A5: Total Monthly Income

D5: + C3 + C4

A7: Mortgage

C7: + 800

A8: 2nd Mortgage

C8: + 200

A9: Auto Loans

C9: + 600

A10: Insurance

C10: + 80

A11: Fuel

C11: + 120

A12: Utilities

C12: + 150

A13: Phone

C13: + 35

A14: Food

C14: + 400

A15: Clothes

C14: + 250

A15: Savings
C15: + D5*.1
A16: Retirement
C16: + C15
A17: Total Expenses
D17: @SUM(C7..C16)
A19: Discretionary Income
D19: + D5 - D17

Now I'm going to describe the "bells and whistles" that make RUN Calc 128 serviceable and more flexible for individual needs.

MENUS AND HELP

As I mentioned earlier, pressing the help key displays a Help screen that describes the functions of keys and the RUN Calc 128 syntax. This Help screen also appears whenever you boot up the program. The escape key is used to exit the worksheet to the Main menu (the slash key also performs this function), or to exit subsequent menus or input routines to get back to your worksheet.

Menus and input prompts are always displayed in the top three rows of the screen. Menu selections are made by pressing the key corresponding to the first letter of the selection, shown in reversed light gray.

Prompts requiring an input are accompanied by a flashing cursor. Press return after responding to any prompt. You'll notice that the Status Flag in the upper-right corner of the screen displays ENTER any time you're dealing with menus. Worksheet operations always work on the column or row where the cursor is positioned.

MAIN MENU

Pressing the escape or slash key

when working on your worksheet displays the Main menu. From the Main menu, you can go on to other menus or quit the program by pressing Q. Any option that will destroy your worksheet will display an "Are you sure?" prompt.

Pressing C clears, or erases, all the cells in the entire worksheet, leaving the column widths as you last defined them. To start over completely, with the default column values, you must quit the program and run it again.

P prompts you for the range, or block, of cells to be printed. The current cell coordinates are preprinted in the prompt. Enter the starting cell coordinates and the ending cell coordinates in the form From..To, as in A1..F20. The starting, or From, cell is in the upper-left corner of the block, and the ending, or To, cell is in the lower-right corner of the block.

This arrangement will appear consistently in other routines, also. The borders shown on the screen aren't printed.

Set up your printer before running RUN Calc 128, because the program isn't capable of sending printer commands after it's activated. RUN Calc 128 prints the worksheet exactly as it appears in memory. If the character width of your chosen block is greater than the character width of your printer, then the rows wrap around and use extra lines on the paper. If your worksheet exceeds 80 characters in width, then you must either put your printer in a compressed mode or print your worksheet in several blocks or sessions. RUN Calc 128 also ignores page perforations, so if you

don't want your rows to be printed on multiple continuous sheets, either limit the size of the block or increase the number of lines per inch on your printer.

RUN Calc 128 sends straight ASCII text, using Basic's PRINT#4,USING token, and it works with any Commodore or compatible printer. Please note that 4 is the default device number for the printer. You can change this address at your leisure right in the program. Graphics characters can only be produced on printers with Commodore graphics.

Pressing W at the Main menu sends you to the Worksheet menu, where you can define and modify the layout of your worksheet.

Pressing F at the Main menu transfers you to the Disk Files menu, where you can perform all operations related to the disk drive. Note that RUN Calc 128 only uses drive device 8. Make appropriate changes to the program if this doesn't meet your needs.

WORKSHEET MENU

The Worksheet menu is where you can alter the width and print format of the columns, as well as delete, insert and erase data selectively. Press escape to exit this and all other menus.

Pressing F at the Worksheet menu sends you to an additional *Format Column menu*, where you can specify how you want the data printed in the cells to be justified.

Pressing W at the Worksheet menu produces a prompt asking for the *width* of the current column. The current column is the column where the

cursor was last positioned before you accessed the menus. The current width of the column will be preprinted for you. You can enter a width of 1-74 characters, and some column formats may require a width of at least four characters.

After you press return, the column and screen are reprinted with the necessary adjustments. Column formats return to the default settings after a width adjustment, so you have to reset the format at the Format Column menu.

When you press D at the Worksheet menu, you can *delete* either the column or row that the cursor is currently positioned in. Press C for column, R for row or escape to exit. When a column is deleted, all columns and the data in them are shifted to the left (column E becomes D, and so forth). Deleted rows make data below the current row shift up a row (row 100 becomes 99, and so forth). RUN Calc 128 automatically adjusts the cell coordinates if @ functions occur in a cell, but normal formulas and functions such as +E1*D4 need to be adjusted manually.

If I, for *Insert*, is pressed at the Worksheet menu, followed by C or R, either a column or row is inserted to the left of or above the current cursor position. Columns to the right of the insert shift right (E becomes F, and so forth), and rows below the insert shift down (99 becomes 100, and so forth). Cell coordinates within formulas may have to be adjusted accordingly.

Erase is a selective way of removing data from cells, as opposed

to Clear at the Main menu. Pressing *E* at the Worksheet menu displays the Erase menu. Pressing *C* or *R* from that menu erases the data within the current column or row. The column width and format remain unchanged.

One additional option, called *Global Range*, lets you erase the data within a block of cells. Choose *Global Range* by pressing *G*, followed by the From..To cell coordinates. Again, the coordinates of the current cell are preprinted as the first cell coordinates. If you want, you can overwrite them with other coordinates. You can also press escape at any time to exit safely without affecting your data.

FORMAT COLUMN MENU

The selections in this menu determine how the data prints within the cells on the screen and in printouts. There is also an option for switching the data type from labels to value, and vice versa. Individual cells can't be formatted separately from the rest of the column. However, labels normally ignore formats intended for values.

The default values are left-justified for labels and right-justified with two decimal places for values. Changing the width of a column normally resets the format for that column to the default values. A cell fills with asterisks (******) if, for some reason, you improperly select a numeric format for the type of data or number of digits that's to be printed within the cell.

Press *L* and your label data will be *left-justified* in the cell in the cur-

rent column. Values continue to be right-justified.

Press *R* and both labels and values become right-justified in the cells in the current column.

If *C*, for *center*, is pressed, labels are centered in the cells in the current column. Values are still flush on the right.

When you press the *D* key, you're prompted to choose how many *decimal* places you want in values. Make sure the column is wide enough to contain the numbers that appear to the left and to the right of the decimal point.

Pressing *M* provides right justification with two decimal places, like the default setting for values, with the exception that a dollar sign (\$) is printed directly to the left of the value, as in \$125.50. Use this selection for currency.

Scientific notation, also called engineering notation, is a popular format for expressing very large or very small numbers that's used by (who else?) scientists and engineers. As an example, this option would print .001234 as $-1.234E-3$. Specify scientific notation by selecting *S*.

Pressing *T* provides two options for *type* of data. An *L* converts data in the column to a *label* by removing the plus or minus sign at the beginning of each cell. The *@* functions aren't affected by switching data types. *V* converts labels to *values* by appending a plus sign to the beginning of the cell data. You'll see the data switch the way it's justified once the screen is reprinted. This option was included to help you import Da-

BASIC LANGUAGE ARITHMETIC OPERATIONS

Symbol	Operation	Examples
+	Add	+ A1 + B2, + 100 + Z250, + H7 + H8 + H9
-	Subtract	+ A1 - B2, - B2 + 35, + C12 - C11 - C10
*	Multiply	+ A1*B2, + A1*(- B2), + 10*4
/	Divide	+ A1/B2, + A1/- B2, - 10/2
↑	Exponent	+ A1↑B2, + 2↑2, + G16↑ - 3

BASIC LANGUAGE LOGICAL OPERATIONS

Symbol	Operation	Examples
=	Equal to	+ A1 = B1, + A1 = 100
<	Less than	+ A1 < B1, + A1 < 100
>	Greater than	+ A1 > B1, - 20 > - 30
<=	Less than or equal to	+ A1 <= B1, + X5 <= X6
>=	Greater than or equal to	+ A1 >= B2, + 1 >= F15
<>	Not equal to	+ A1 <> B2, + 100 <> J23

Table 1. RUN Calc 128 operations and functions.

tafile records into RUN Calc 128 by letting you select specific fields (which become columns) to be converted from strings (or labels) to values.

DISK FILES MENU

RUN Calc 128 works with disk device 8, so keep your data disk in that drive. Also, store a copy of the program on the same disk as your worksheets. Worksheets are saved on the disk with a 12-character name

of your choosing, preceded by cf]. When typing in worksheet filenames, don't type in these special characters. Any time the program is accessing the disk drive, the status line displays a flashing DISK.

Pressing D clears the worksheet and displays a *directory* of all files on the disk. Pressing escape exits the directory listing and redisplay your worksheet on the screen.

Pressing L clears the worksheet, prints only RUN Calc 128 work-

BASIC LANGUAGE FUNCTIONS

Symbol	Function	Examples
ABS ()	Absolute	+ ABS(A1), + A1 + ABS(B1 = B2)
ATN ()	Arctangent	+ ATN(A1), + 2*ATN(2*3.14159)
COS ()	Cosine	+ COS(A1), + ATN(COS(2*3.14159))
EXP ()	Exponent	+ EXP(A1), + 25*EXP(X1)
FRE ()	Free memory	+ FRE(0)
INT ()	Integer	+ INT(A1), + INT(A1*100 + .5)/100
LOG ()	Logarithm	+ LOG(A1), + EXP(A1)/log(A1)
RND ()	Random number	+ RND(1), + INT(RND(1)*6 + 1)
SGN ()	Sign	+ SGN(A1), + A1*SGN(B1 - C1)
SIN ()	Sine	+ SIN(A1), + 2*SIN(30*2*3.14159)
SQR ()	Square root	+ SQR(A1), + SQR(A112)
TAN ()	Tangent	+ TAN(A1), + TAN(SIN(A1)/COS(A1))

@ FUNCTIONS

Symbol	Function
@SUM(A1..A10)	Sum, or add, the values in a range of cells.
@AVG(A1..Z1)	Find the average value for a range of cells.
@MAX(A1..Z250)	Find the maximum value in a range of cells.
@MIN(Z1..Z50)	Find the minimum value in a range of cells.

sheets (those sequential files beginning with cf1) listed in the directory and initiates a prompt asking you to enter the name of the worksheet file. Then, when you press return, the worksheet file is loaded into the cells from which they were originally saved. Loading a worksheet file doesn't clear the current worksheet, which means you can combine several sheets, as long as they don't occupy the same cells. You can move the cells of the current work-

sheet by inserting the appropriate number of columns or rows to make room for the incoming worksheet file. The last worksheet file loaded determines the column format settings for the entire worksheet.

When you press S at the Disk Files menu to save the current worksheet, a prompt appears asking for the worksheet filename. Enter a filename that's 12 characters or less and hasn't been used before on the disk. If you want to reuse a filename, you

can rename and erase the old worksheet file on the disk. Worksheets are saved to disk with the cell coordinates to make sure they'll be placed back in the same cells upon loading. Consequently, you must input the range, or block, of cells to be saved, using the From..To format.

To format a disk, press *H*, for *header*, and you'll be asked for a 16-character disk name and a two-character disk ID. If the disk has already been headed, you can dispense with the ID by pressing return without an entry. Press escape to exit at any time.

At some time, you may want to *rename* a worksheet file so you can save your current worksheet using an existing name. Press *R* to have the program display a listing of existing RUN Calc 128 worksheets on the disk for your examination. Enter the name you want changed after the From prompt and the new name after the To prompt.

Erase is another term for *scratch*, which means to delete excess files on your disk. Pressing *E* displays a directory of RUN Calc 128 worksheets and a prompt for the name of the file to be erased. Erasing is more or less permanent, so be sure you really won't need the worksheet anymore.

Pressing *I* at the Disk Files menu accesses a routine that reads and *imports* a sequential record file created by Datafile 3.6 (available on ReRUN Productivity Pak II, and January-February and March-April 1987 ReRUNs). RUN Calc 128 analyzes the structure of the record file, then

begins loading the records into corresponding rows. Each column's width is set according to the width of the fields.

The record's file data is brought in as labels, and you're limited to 250 records using 26 fields (the maximum spreadsheet size). You can convert numeric string fields to values and select column formats by going to the Format Column menu at a later time.

If you have more than 250 records or 26 fields in your record file, you must make a subset of the file using DFC1one (ReRUN, July-August 1987). Keep in mind that you may need extra rows in which to place titles and equations for totaling your data. You may want a few extra columns also, but in most cases you'll probably delete unnecessary columns holding fields that aren't needed. RUN Calc 128 produces a more customized report than is possible with the database programs themselves.

Place your database record file disk in the drive. Then, when you press *I* at the Disk Files menu, RUN Calc 128 displays a directory of compatible record files. Datafile record filenames are preceded by *df*]. Type in the name of the record file you want and the program will load it and convert it into a worksheet. The records are transferred in the same sorted order they have in the record files, so perform any sorting operations before importing. Once the Datafile files are imported, they can't be exported back to the databases. ■

RUN Calc 64

By Trent Busch

RUN It Right

C-64

RUN Calc 64, an electronic spreadsheet program for the C-64, can be used for calculations that range from checkbook balancing to complex investment analysis. This article takes you step by step through the program's features.

RUN Calc 64 appears on the Productivity Pak III disk in two forms. One, RUN Calc 64, is a compiled program that offers maximum speed. The other is the original Basic program, RUN Calc 64.bas, which we've included for those who want to see how the program works.

VIEWING THE SPREADSHEET

After loading and running RUN Calc 64, examine the display. The flashing cursor at the top-left of the screen is on the data-entry line. Running across the screen below it is a solid comment line that the program uses to display important information. The initial screen also displays the numbers 0, 1 and 2, which represent columns, and the letters A-T, which identify the rows. Each of the columns is limited to holding a maximum

imum of nine characters.

RUN Calc 64 has a total of 30 columns and 26 rows, but only three columns and 20 rows are visible on the screen at one time. It's like looking through a window that reveals only a portion of the landscape outside.

The cursor keys let you move this window around the spreadsheet. Press the cursor-down key, and the quickly redrawn display shows you rows B-U. Experiment with the cursor keys to move the viewing window over all the columns and rows.

ENTERING DATA

The intersection of a spreadsheet column and row is called a cell, designated by the row letter and column number. RUN Calc 64 provides 780 cells: A0-Z29. Pressing the home key at any time will return the window to the initial A0 position.

To enter information into a cell, follow this simple procedure. First, type in the cell location, with the row first, the column second, and no spaces between the characters. Next, type a colon to separate the cell location from the data. Then type in your text or numeric data up to nine characters in length. The nine characters include control codes,

which each count as one. As examples of cell input,

A0:BUDGET 88

would enter a label in cell A0, and

C12:250

would enter the value 250 in cell C12.

Text information can include almost any character, but mustn't begin with a number or a plus or minus sign. You can highlight text information by inserting color changes or reverse imaging, but be sure to change the color back to normal afterwards. Numeric information *must* begin with a number or a plus or minus sign.

After typing the information you want in a cell, press the return key. If you typed everything correctly, the data should appear in the proper cell.

It's very important that you enter information by the proper method. If you don't, RUN Calc 64 displays an error message on the comment line to help you locate the problem. When you're ready to enter the line again, press any key and just retype the line correctly.

Numeric data is automatically aligned flush right in a cell, and text data is aligned flush left. You can insert spaces to move the text over if you want.

To clear a cell, enter the cell coordinates, followed by a colon, and then press return. This procedure will clear text, but not formulas. To clear the entire spreadsheet, press shift/clear and answer Y to the question on the comment line.

ENTERING FORMULAS

Up to now, all you've done is create neat columns and rows. The real power of RUN Calc 64 lies in its ability to perform mathematical operations using the data in the cells. For example, you can add cell A0 to cell A1 and place the answer in cell A2 by putting the formula $A1 + A2$ in cell A2. Here's the proper format:

A2:[F1]A1 + A2

The F1 appears as a reverse F and is printed by pressing the F1 key (don't type the brackets). This key accesses special features of RUN Calc 64. If you forget to press F1 when entering a formula, it will be displayed in the cell as text. If you do press F1, the result of the computation will appear in the cell.

When typing in a formula, leave out all spaces and enter the cell first and the formula second.

A special View command lets you view the formula in a particular cell. This is the only way to view formulas that are more than nine characters long. As an example of this command, if you enter:

A2:[F1]V

and there's a formula in cell A2, it'll be printed on the comment line, along with the full value of the numeric data in the cell.

RUN Calc 64 fills a cell with asterisks when it contains a formula that's more than nine characters long. The asterisks alert you to use this View command to examine the contents of the cell.

DOING ARITHMETIC

RUN Calc 64 can perform the following mathematical operations:

- Addition: cell + cell or cell + constant
- Subtraction: cell - cell or cell - constant
- Multiplication: cell * cell or cell * constant
- Division: cell / cell or cell / constant
- Exponentiation: cell ^ cell or cell ^ constant

The program doesn't handle complex formulas, but you can do involved computations by storing an intermediate answer in a spare cell.

Calculations aren't made immediately as data is entered, but are triggered by your pressing the left-arrow key at the top-left of the keyboard. The calculations take a while, depending on the number of formulas in the spreadsheet. They're performed column by column, from top to bottom, with one column completely calculated before the next one is started.

Sometimes this can result in calculations being done in the wrong sequence. For example, say cell A0 contains F9 * G6. If cell F9 has a formula in it, the results of that formula will not be calculated until *after* cell A0 is computed. To get around this problem, press the left-arrow key twice.

After the program has completed all computations, the spreadsheet is redrawn, with the results displayed in the proper cells. An error message will appear if you attempt to divide by zero, and, if a result of exponentiation is too large, an overflow note

will appear in the affected cell.

SPECIAL FORMULAS

RUN Calc 64 provides the following special formulas. They're presented here with sample data inserted.

A1:[F1]SUMA2-Z2 places the sum of cells A2 through Z2 in cell A1.

Z29:[F1]AVGB3-B12 calculates the average of cells B3 through B12 and places the answer in cell Z29.

C12:[F1]MIND0-G0 searches for the minimum number in cells D0 through G0 and places the answer in cell C12.

F5:[F1]MAXZ0-Z29 returns the maximum value in a range.

These formulas work only on a range of cells in one row or in one column. They will not work on diagonal groupings of cells.

SPECIAL COMMANDS

RUN Calc 64 also provides the following special commands, which are again presented with sample data.

A0:[F1]T leaves row A and column 0 titles displayed on the screen, so you can see what each cell is supposed to be as you move the window around. You must use cell A0 in this command.

A0:[F1]O turns off the title mode. The cell must be A0.

C15:[F1]A clears a single cell of formula, text and numeric data.

F25:[F1]J jumps the display window to a particular area of the spreadsheet. Sometimes this is faster and easier than using the cursor keys.

D3:[F1]COPD4-D29 copies the contents of a cell, which is handy

when you have to enter lots of identical information. In this example, the contents of cell D3 will be copied into cells D4 through D29. Only text and numeric information can be copied in this way; formulas must be typed in individually. This command works with rows or columns.

THE FUNCTION KEYS

The role of F1 in entering formulas has already been discussed.

Pressing F2 shows you formula results on the comment line to a degree of accuracy up to six decimal places. RUN Calc 64 automatically rounds formula results to two decimal places. To display a different degree of accuracy, press F2 and then a number from 0 to 6 to indicate the number of decimal places you want. This command doesn't affect the accuracy of the calculations, and you can always use the View command to see the actual value. As mentioned, the precision setting pertains only to numbers that are the results of formulas. Other numeric data must be typed in just as you want it.

Press F3 and follow the screen directions to load the spreadsheet template from disk.

Press F4 and follow the screen directions to save the template to

disk. Be sure to pick a filename that suggests the subject of the data the template holds.

Press F5 and follow the screen directions to print the spreadsheet template on paper. You can print the whole template, or you can print a portion of it by entering the coordinates of the top-left and bottom-right cells of the part you want. If you specify more than seven columns, RUN Calc 64 will automatically break the printout into sections.

Press F6 and follow the screen directions to print the formulas in the template on paper.

Press F7 to change the background screen color, F8 to change the border color and the control key with a number key, 1-8, to change the text color. After you enter any color change, press return; ignore the error message and press return again.

The possible applications of RUN Calc 64 are innumerable. To start with, you might want to try the template in the November 1986 issue of *RUN* (p. 40). The December 1986 issue of *RUN* contains, on p. 82, a nice checkbook-balancing template, illustrating another practical use. The more you use RUN Calc 64, the more applications you're sure to find for it. ■

RUN Notepad 128

By Bob Kodadek

RUN It Right

C-128

The 128 Notepad program is a commercial-quality desktop accessory for your C-128 that provides many functions. Above all, it offers a text window for recording and recalling programming and other notes, but it also includes a digital time display; a menu for viewing the disk directory and for saving, loading and printing your Notepad notes; and a DOS command center for file maintenance and other disk operations.

Once 128 Notepad is activated, you can interrupt the program you're running, open the window and enter the text editor by pressing control/W. In turn, control/V exits 128 Notepad and returns you to the original program, where the screen display will be exactly the same as you left it, right down to the cursor location.

Files produced with the 64 Notepad Update from last January's issue of *RUN* are compatible with 128 Notepad in the C-128's 80-Column mode.

The 128 Notepad program is written entirely in machine language, resides completely in RAM and remains transparent to most other pro-

grams. You can use it in Direct mode as a stand-alone application, with the C-128's machine language monitor or with any Basic or machine language program, including RUN Script 128 Version 2.40 (*RUN*, December 1986, and ReRUN Productivity Pak II) and the Time-DOS 64/128 date-and-time-stamp utility from Family Software. When 128 Notepad is operating with Time-DOS 128 and a Basic program, you'll have three programs interfacing at once on your C-128!

The disk drive's error channel is continually monitored while 128 Notepad is running, and its status is displayed after every operation.

LOADING

The 128 Notepad program actually consists of two listings: Boot 128 Notepad and ML File Generator. A fast-loading binary file appears in the disk directory as 128 NOTEPAD.OBJ. To activate it, you must run 128 Notepad Boot. The boot program loads the object file and then asks you to enter the current time in hours and minutes. If you press the return key without a numeric entry at these two prompts, the TOD (time-of-day) clock will be set to zero, so it functions as a timer.

If you have Time-DOS 128 in memory, the time is set automatically, and all you need to do is activate the object file. If you own a 1571 disk drive, type BOOT "128 NOTEPAD.OBJ" and press return; if your drive is a 1541 or compatible, type BLOAD"128 NOTE PAD.OBJ", press return, type SYS 7181 and press return again.

Owners of a CCSZ clock/calendar card from the Jason-Ranheim Company must permanently set the TOD master to CIA #2 and disable the time tick, in addition to inputting the Boot or BLoad command above. In most instances, I've found it best to turn off the CCSZ when powering up by typing in POKE 165,30:SYS 999. The TOD clock will continue to keep the correct time, even after you disable the card.

Since 128 Notepad relocates the start of Basic to memory address \$4000, you must always input the New command before loading a Basic program when Notepad is in use. New resets all the Basic pointers.

USING THE NOTEPAD

Once you've activated 128 Notepad, press control/W to open the window and access the text editor. As the window opens, it will display a Help screen. To clear the window and go on to typing in your notes, press the shifted clear-home key.

All of the C-128's screen-editing features, including the Escape command sequences, function within the window. Insert/delete, the cursor controls and the extended keyboard also operate normally, effectively

making 128 Notepad a mini word processor. The program provides 562 bytes of text storage, consisting of 14 lines of 38 spaces each and a 15th line of 30 spaces. The lower-right corner of the window always displays the current time.

When you're done using the window, close it by pressing control/V from the text editor. If you want to leave 128 Notepad altogether, press run-stop/restore. You can reactivate it by inputting SYS 7181.

If you ever want to open the Notepad window while the screen is scrolling, you *must* first stop the scrolling with the no-scroll key or control/S, or the registers in the 128's 8563 80-column video chip will be confused. If it should become necessary to reset the chip, close the Notepad window and press run-stop/restore.

THE OPTION MENU

To access the 128 Notepad option menu, press control/D while the window is open. The menu choices are listed in Table 1.

Pressing D displays the disk directory. While the directory is scrolling, you can pause it with the no-scroll key or halt it altogether with the stop key.

When you press S to save or L to load a Notepad file, the prompt SAVE:N. or LOAD:N. will appear. The N. is a prefix that 128 Notepad automatically adds to the names of its files. The program's custom input routine will accept any filename that's no longer than 14 characters. To abort a save or load, press the

return key without entering a filename, or press the stop key.

When you press P to print out a window, only the Notepad border and its text are output, so it's easy to fit several hard copies on a single sheet of paper.

Pressing X accesses disk operations, including copying, renaming and scratching files; initializing and resetting the drive; and validating and formatting disks. When the prompt for the command string appears, respond with the standard format for the command you want. For example, to rename a file, you'd input R0:newname=oldname, and to scratch a file, you'd input S0:filename. Filenames must include the N. prefix.

If you own a 1571 drive, you can try sending User 0 commands from the Notepad window. These commands aren't mentioned in the user's manual for the drive, but they're quite useful. A list of some of the more interesting ones appears in Table 2. The format for User 0 commands is U0>xx, where xx is the operation you want.

The last command listed in Table 2 will keep your drive head from banging. The x is the number of read attempts in the drive's zero page location, \$6A. Try U0>R1.

The last choice in the option menu is Q, which exits from the menu and takes you back to the text editor. If you want to go further and close the window, press control/V.

THE NOTEPAD AND RUN SCRIPT

To use 128 Notepad with RUN

Script 128, first load and run the Notepad boot program, then load and run the RUN Script 128 boot program. *Never press the restore key with the window open when you're using the Notepad with RUN Script 128.* If you do, the computer will lock up, because RUN Script alters the NMI (nonmaskable interrupt) vector, and you'll have to reset the computer.

After you've activated 128 Notepad, the interrupt waits for a control/W. When it detects one, a flag diverts all future interrupts, and the current screen display (including attribute memory) is transferred to an alternate location at address \$1000 in the 8563 chip's RAM. This chip comes with its own block-move routine, so the process is easy.

Once 128 Notepad has saved the contents of the zero page and the 8563's VDC (video display controller) registers, it switches in an alternate screen that exactly duplicates the original. The Notepad window will appear in this duplicate. When the alternate screen is in place, the program clears the interrupt-disable bit. Specific Notepad subroutines perform all operations on the new screen.

Once you close the window, the program switches back to the original screen by reinstating the contents of the zero page and the VDC registers that were stored. It then clears the flag and jumps to \$FF33 to return control to Basic or the program in memory.

The official *Commodore 128 Programmer's Reference Guide* con-

tains almost all the information you'll need to program the 8563 chip. However, it fails to mention that free RAM at \$1000-\$1FFF can hold an alternate screen and machine language routines for reading and writing to the 8563's VDC registers from Basic.

These ML routines exist in the screen editor ROM and appear in bank 15 at \$CDCC and \$CDDA, respectively.

To write to a 8563 VDC register from Basic 7.0, use the format `SYS DEC("CDCC"),A,X`. The X register holds the number of the VDC register to be accessed, while the A register contains the value to be stored in it. For example, to change to the alternate starting address of display RAM,

write the value 16 to VDC register 12 by using the code `A=16:X=12:SYS DEC("CDCC"),A,X`.

To read a register, load the X register with the number of the VDC register to be read from and call the routine at \$CDDA. The routine will return to Basic with the value in the A register. To change the value into a Basic variable, use the `RREG` command. For example, to read VDC register 12 in Basic, the code would be `X=12:SYS DEC("CDDA"),A,X:RREG A`. The variable A will equal the value found, in this case 16.

RUN readers have been enthusiastic about the 64 Notepad program (September 1986 issue). I think you'll be just as pleased with 128 Notepad. ■

RUN Notepad 64

By Bob Kodadek

RUN It Right

C-64

While computing, how often do you have to spend time looking up Poke, Peek or SYS numbers because you forgot them? Or how often have you had an idea you wanted to write down quickly but were unable to find a pencil or paper handy? If you're like me, you end up making a lot of mental notes, then forgetting them.

With RUN Notepad 64, you have an electronic pad, pencil and eraser at your fingertips.

ABOUT THE PROGRAM

After you run this program, you'll be asked to enter the correct time. The first input prompt will request hours, and a second prompt, minutes. Since this is a time-of-day clock, the hours must be any number from 1 through 12. By pressing only the return key in response to these two prompts, the internal clock will start at zero and may be used as a timer. The current time is always displayed in the lower-right corner of the text window. Although the digital time display stops during printing, the clock will still keep the correct

time. Consequently, the display is updated automatically when the printing operation is complete.

After the Ready prompt appears, you can use the Notepad. Press the CTRL/O combination to open the window. The routines will take over the function keys whenever the window is open, and they will perform the following functions:

- F1:** Saves a Notepad window.
- F3:** Prints the window currently in memory.
- F5:** Prints a full screen.
- F7:** Loads a Notepad window.

Press CTRL/C to close the window, and the keys should revert to their original functions.

When you wish to save (F1) or load (F7) a window, one of the prompts, SAVE: N. or LOAD: N., will appear. Make sure a formatted word disk is in the drive, then give the program a filename. Your notes will be safely stored, and you'll have a fresh workscreen to write in. To abort at any time, either press the stop key or press return without entering a filename.

The N. in the prompts is a Notepad file identifier, and it automatically becomes part of the filename. There's no need to type in this prefix; it is

done for you. Remember that you are limited to 14 characters in the filename because of the prefix. The Custom Input routine, complete with cursor and delete functions, also limits your input to 14 characters.

Each time you save or load a window, the error channel is read and displayed on the screen. Press return,

D	Directory
N	Format
R	Rename
I	Initialize
B	BLoad
V	Validate
C	Copy
S	Scratch
Q	Quit

Table 1. Notepad DOS commands.

and your text will appear. These commands also make it possible to use RUN Notepad 64 as a simple index card file. A single-sided disk can hold up to 144 Notepad files and is only limited by the directory.

USING THE DOS ROUTINES

You can access the DOS menu whenever the Notepad window is open by pressing control/D. The one-key commands that will appear are listed in Table 1. To select the function you want, just press the appropriate key. If the routine needs any information from you, it will issue a prompt at the appropriate time.

You can abort any operation on the menu by pressing the stop key, and the current status of the error channel always appears at the bottom of the menu screen. To return to viewing your notes, select the Exit option, or, to return to Basic, press control/C to close the window. ■

RUN Dex

By Barbara Mintz

RUN It Right

C-64

If you'd like to set up a simple, versatile file system with a minimum of effort and a maximum of flexibility, then it's time to tuck the RUN Memo Book into your C-64. Memo Book is an electronic "Rolodex" you can organize and format any way you want, then flip through the pages at will. And "flip" it is, for the machine language program is fast, and your data is held in memory, so no time is lost in disk access.

During initialization, Memo Book installs itself at the top of Basic memory, starting at address \$9400. Then it resets the top-of-Basic pointers to that address, so Basic programs won't invade Memo Book's space. About 141 blocks are left free for a second program. Memo Book also inserts the address of its own machine code in Basic's error vector at \$300, to intercept the routine that prints Basic's error messages. This makes Memo Book's index page, instead of the usual Syntax Error message, appear each time you press * and return.

When initialization is complete, a subroutine clears the Basic program

area, resets all the pointers and displays Memo Book's index page. You can go on to set up your Memo Book at this point, or, if it's already set up, you can leave the program by pressing Q and return, and load another program. Unless the program you load is unusually long or changes the Basic pointers (which many commercial programs do), you can continue to read and write memos until you turn off your computer.

If a crash forces a reset, you can probably reinitialize Memo Book with SYS 40486. Even if you can't, your entries should be safe, because they're stored out of the way under Basic's ROM. Nevertheless, when you've made a lot of entries, better safe than sorry—back up your data by saving it to disk.

GETTING STARTED

When you use the program for the first time, you'll have to set up your memo book. To start this process, press F2. Then, at the question "Save This Book?", respond with N, since you don't yet have a book to save. Finally, press return, and the first empty page of your memo book will appear.

If you've already set up a book with data and saved it to disk, this

is the time to load it into memory. Press F6 and, at the prompt, type the filename of the book and press return. When the index page appears, use the cursor keys to scroll to the page you want, then press return to display the page on the screen. You can also "flip" through the pages of your book by scrolling. Use the cursor keys to scroll rapidly, or F1 and F7 to scroll slowly. If the number of entries makes scrolling through the pages unwieldy, press return to go back to the index and pop to another page.

Memo Book has relatively few commands to remember, and I've kept those few as intuitive as possible. If you do need to refresh your memory, enter ? to see the Help screen. Also, if you ever issue the wrong command, you can abort it with a return. Commands that affect the entire book must be entered from the index page.

INPUTTING ENTRIES

The command for adding entries to your memo book is control/I (for insert). It works whenever the cursor points to an empty line, whether you're starting a new book or adding entries to one that's already established.

After you press control/I, the screen clears, and a red prompt message instructs you to input an entry. The entry can be up to six lines long and 38 columns wide, and you can format it any way you like, with whatever punctuation you find convenient. If one six-line entry isn't enough, continue right below with

another. If you have trouble staying within the 38-column limit, the segmented guideline I've provided should help.

Press return at the end of each line, and signal completion of your entry by pressing return on an empty line. You'll find yourself back in the memo book, looking at the new entry. When you're adding a lot of new entries, it's wise to save your data frequently.

EDITING ENTRIES

To edit an entry, press control/E. Once again the screen will clear, but this time the entry you want to edit will appear, with a flashing cursor at its first character. Edit one line of the entry at a time, using all the normal keys to insert, delete and overwrite characters. When you finish a line, press return. In fact, you must press return for each line you want to retain, whether you change it or not. Conversely, if you want to eliminate a line, move the cursor to the next line without pressing return.

To insert a new line in an entry, move the cursor to an empty line, type in your data and press return. Then move the cursor back to edit the next line.

When you've finished editing your entry, press return on an empty line to put it back in the memo book.

DELETING ENTRIES

Control/D, as you might expect, deletes an entry. To prevent accidents, you must confirm that the entry, now highlighted in red, really should be deleted. If you respond

Y, the entry will disappear from the screen and from your memo book.

SAVING A BOOK

Press F4 to save a book. When the program asks if you want to update a file, a yes response will scratch the old file, and the book you're currently using will be saved in its place. As I've mentioned, F2 starts a completely new book. Here also, you have the option of saving the current book first.

A full book occupies about 35 blocks in memory locations \$A000-\$C000. However, only the portion that actually contains data is saved and reloaded, so no time and disk space is wasted on empty bytes.

PRINTING

To print out one entry, press control/P, then, at the prompts, choose to print the entire entry, omit the last line or omit the last two lines. If you keep phone numbers and comments on those lines, you'll want to eliminate them when you print addresses on envelopes or labels.

To print a full page of entries, press F3. Again, you can choose to eliminate lines, and the page will be printed from the cursor position. F5 prints the entire book, with the same options.

RETURNING TO BASIC

When you're done consulting your memo book, if you're on a memo

page, press control/Q to quit and return to Basic. If you're on the index page, Q alone will do the same thing. You can jump back to Memo Book from Basic at any time by pressing * and return in Direct mode.

MULTIPLE APPLICATIONS

Because it's so unstructured, Memo Book can be used for a variety of purposes simultaneously. It's easy, for example, to add appointments and a list of "things to do" to the top of the first page, under A. If you don't scroll the index, the book will automatically open there, for a quick check of your schedule.

It's equally easy to set up and use several books for different purposes, because Memo Book lets you save one book and load another by just returning to the index and inputting F6. Because the load will obliterate the data you've been working on, the program will ask if you want to save the present book before loading the next one. If you respond with a Y, you'll go through a save routine similar to the one I described earlier.

When that book has been taken care of, you'll be prompted for the name of the book you want to load, and the load will take place. It's a good idea to keep a list of your memo books somewhere in each book, so you don't have to consult the disk directory for the correct names. ■

RUN Investor

By Stan Krawczyk

RUN It Right

C128

I wrote RUN Investor, a C-128, 80-Column mode program, to keep track of my stocks on a daily basis. It tells me how much each of my stocks is worth at its current price and whether I'm making or losing money on it. For instance, say I buy XYZ Co. stock for \$10 a share in June, \$11 a share in July and \$9.50 a share in September. RUN Investor will tell me the value of each transaction if the stock is currently selling for \$10.50 a share, as well as the total market value at that price.

RUN Investor supports one or two disk drives (devices 8 and 9) and the Commodore 1525/MPS-803 or 1526-compatible printer. It's menu-driven and easy to operate, and it uses the C-128's window capabilities.

I've written the program in logical sections, so you'll find it easy to locate the code for different functions. For example, the List File option, which is activated by pressing F3, is located in lines 300-400 and 1300-1400 and the Update Stock Price option, F7, is located in lines 700-800.

Note that the function keys, F1-F8,

are redefined in lines 5 and 6 to their C-64 values. The number of disk drives is determined automatically in line 7. If the inquiry to device 9 results in a status of 0, the program sets itself up for two-drive operation. Otherwise, it uses just one.

As written, RUN Investor accommodates up to 100 stock names. However, you can increase this number as your needs grow by changing the DIM values in line 4.

CREATING A STOCK FILE

To use RUN Investor, key in the date and initialize a stock file by pressing the F5 key. Then enter, in turn, the name of the company whose stock you want to buy or already own, the number of shares involved—stored in variable SO(I)—and the cost per share. The number of shares can be fractional, down to a ten-thousandth of a share—for instance, 7.1275 shares. When entering the cost per share, use multiples of .125 to represent eighths of a share. You can write your data to disk (device 8 or 9, or both) as soon as you've finished entering it, or you can wait until later. Press F8 to execute the save.

To see the total market value of your stock, press F3. You can print

out a report from this option, too.

To create another file, perhaps for another member of your family or to follow some stocks you may want to buy later, press F5 again. Just remember to save it to another disk and to label it accordingly. The program writes the filename Stock File to disk, so make sure your label indicates which stock file. Any previous stock file on a disk will be overwritten when a second save is done.

To quit the program, press the F6 key. If you haven't saved your file, the program will remind you, so the file won't be lost.

The upper-left portion of the RUN Investor screen displays the program and error status. This is done with a POKE 235,0 and PRINT"", which eliminates any character flash on the screen and places the next print on line 1 of the screen.

UPDATING YOUR FILE

The fun begins when your daily paper is delivered and it's time to see how your investments are doing. With the stock section of the paper in hand, load RUN Investor, then enter the date, press F1 to load your file and F7 to update the prices. Your stocks will be listed one at a time, with a prompt for the current quote. If a particular stock hasn't changed in value, enter N or hit the return key; if it has changed, enter Y and the new price, then verify the amount when prompted. After you've updated all your stocks, you can save the update to disk, or wait until later. Press the F3 key to see your gain or loss on each issue of stock and

to print out a hard copy.

Say some time lapses and you buy more stock. All you have to do is load RUN Investor, press the F4 key and follow the prompts.

If you decide to sell some stock, press the F2 key and answer the prompts. The program will total all shares of the stock name you want to sell, ask how many shares you plan to let go (in whole amounts or fractions down to a ten-thousandth of a share) and then subtract the number you've specified. When all the shares are gone from one issue, it will go to the next issue, as long as the stock name is the same. When an issue is completely gone, all the variables move up one and automatically delete the entry from the file in memory.

If you have a stock file on a second disk, you can load it by pressing F1. The program will remind you if you already have a file in memory. If you want to go ahead and load the new file, change disks and the loading will proceed.

Right now I use RUN Investor's Stock Purchase option (F4) to record dividends reinvested automatically by my broker. When the program asks for a stock, I enter the number of the stock that paid the dividend, then the number of shares bought (the dividend), a purchase price of \$0.00 (since it's a dividend) and the price the dividend was issued at (from my broker's report). If the program arouses enough interest, I plan to expand RUN Investor to handle dividend reinvestment and tracking and to graph selected periods of a stock. ■

Please send me back issues of ReRUN

- | | |
|---|---|
| <input type="checkbox"/> January/February 1986 | <input type="checkbox"/> November/December 1986 |
| <input type="checkbox"/> March/April 1986 | <input type="checkbox"/> January/February 1987 |
| <input type="checkbox"/> May/June 1986 | <input type="checkbox"/> March/April 1987 |
| <input type="checkbox"/> July/August 1986 | <input type="checkbox"/> May/June 1987 |
| <input type="checkbox"/> Productivity Pak II | <input type="checkbox"/> July/August 1987 |
| <input type="checkbox"/> September/October 1986 | <input type="checkbox"/> Disk version(s) at \$16.47 each* |

* Price includes postage and handling. For foreign air mail, please add U.S. \$1.50 per item.
Prepayment only.

☐ Payment Enclosed ☐ MC ☐ VISA ☐ AE

Card # _____ Exp. Date _____

Name _____

Address _____

City _____ State _____ Zip _____

Signature _____

ReRUN • 80 Elm Street • Peterborough, NH • 03458

BEAT THE RUSH!

Please send me:

- ☐ 1 year (6 issues) for \$69.97
☐ September/October 1987 ReRUN disk for \$16.47.*

**Available in October 1987.*

Includes programs for C-64 and C-128 (in both 64 and 128 modes).

Price includes postage and handling. For foreign air mail, please add U.S. \$1.50 per item and \$25 per subscription. Prepayment only. Subscriptions will start with the current issue.

☐ Payment Enclosed ☐ MC ☐ VISA ☐ AE

Card # _____ Exp. Date _____

Name _____

Address _____

City _____ State _____ Zip _____

Signature _____

ReRUN • 80 Elm Street • Peterborough, NH • 03458

Edit Mode Key Commands

F1-Find F3-First F5-Next F7-New

F2-Write F4-Last F6-Prev F8-Print

Edit Mode Key Commands

F1-Find
F2-Write

F3-First
F4-Last

F5-Next
F6-Prev

F7-New
F8-Print

Keyboard overlays for RUN File 1.0.

Productivity Pak III

For your C-64 or C-128

- **Word Processor ► Spelling Checker ► Dictionary**
- **Database ► Spreadsheet**
- **Electronic Notepad ► Memo Book**

For the C-64:

- **Menu 64**
- **RUN Script**
- **Spelling Checker***
- **Dictionary Maintenance***
- **RUN Calc 64**
- **RUN Dex***
- **RUN Notepad 64**
- **RUN File 64***

For the C-128:

- **Menu 128**
- **RUN Script 128**
- **Spelling Checker***
- **Dictionary Maintenance***
- **RUN Calc 128***
- **RUN Investor***
- **RUN Notepad 128**
- **RUN File 128***

All previously published programs are revised and updated for the Productivity Pak III.

*** Never-before-published programs.**

If any manufacturing defect becomes apparent, the defective disk will be replaced free of charge if returned by prepaid mail within 30 days of purchase. Send it, with a letter specifying the defect, to:

ReRUN • 80 Elm Street • Peterborough, NH 03458

Replacements will not be made if the disk has been altered, repaired or misused through negligence, or if it shows signs of excessive wear or is damaged by equipment.

The programs in ReRUN are taken directly from listings prepared to accompany articles in *RUN* magazine. They will not run under all system configurations. Use the RUN It Right information included with each article as your guide.

The entire contents are copyrighted 1987 by CW Communications/Peterborough. Unauthorized duplication is a violation of applicable laws.

©Copyright 1987 CW Communications/Peterborough



CW COMMUNICATIONS/PETERBOROUGH